## Table of Contents

1. **API Authentication** ................................................................. 22
2. **HTTP Methods** ....................................................................... 23
3. **HTTP Response Codes** .......................................................... 24
4. **Formatting and Naming Conventions** ....................................... 25
5. **FAQ** ...................................................................................... 27
6. **About API Guide** .................................................................... 28
   6.1 API Authentication ............................................................. 28
   6.2 HTTP Methods ........................................................................ 28
   6.3 HTTP Response Codes .......................................................... 28
   6.4 Formatting and Naming Conventions ....................................... 29
   6.5 FAQ .................................................................................... 30
7. **Accelerator** ............................................................................. 32
   7.1 Get List of Accelerated Networks ........................................... 32
   7.2 Enable Network Acceleration ............................................... 33
   7.3 Disable Network Acceleration ............................................... 33
   7.4 Enable Accelerator for Virtual Server .................................... 33
   7.5 Disable Accelerator for Virtual Server .................................... 34
   7.6 Enable Acceleration for Smart Server ..................................... 34
   7.7 Disable Acceleration for Smart Server ..................................... 35
8. **Accelerator Statistics** .............................................................. 36
   8.1 Get Summary Charts ............................................................. 36
   8.2 Get Summary Metrics ............................................................ 37
9. **Add Zabbix Server** ................................................................... 38
10. **Alerts** ................................................................................... 39
11. **Application Servers** .............................................................. 40
   11.1 Create Application Server .................................................... 41
12. **Apps for Application Servers** ................................................ 46
   12.1 Get List of All Installed Applications .................................... 46
   12.2 Get List of All Available for Installation Applications ............ 47
   12.3 Get Application Attributes .................................................. 48
   12.4 Install Application .............................................................. 48
   12.5 Back Up Application ............................................................ 49
   12.6 Delete Application .............................................................. 50
   12.7 Get List of All Application Backups ....................................... 51
   12.8 Restore Application Backup ................................................ 52
   12.9 Remove Application Backup ................................................ 52
   12.10 System Applications ............................................................ 53
    12.10.1 Get List of System Applications ...................................... 53
    12.10.2 Install System Application ............................................. 54
    12.10.3 Switch PHP Version ....................................................... 55
    12.10.4 Uninstall System Application .......................................... 55
   12.11 Domains ............................................................................. 56
   12.11.1 Get List of Domains ......................................................... 56
   12.11.2 Create Domain ............................................................... 56
   12.11.3 Delete Domain ............................................................... 58
   12.12 FTP Users ........................................................................... 59
   12.12.1 Get List of FTP Users ....................................................... 59
   12.12.2 Create FTP User ............................................................ 60
   12.12.3 Delete FTP User ............................................................ 62
   12.13 Databases .......................................................................... 61
   12.13.1 Get List of Databases ....................................................... 61
   12.13.2 Create Database ............................................................ 62
12.13.3 Delete Database ............................................................... 62
12.13.4 Get List of Database Users ............................................. 63
12.13.5 Get List of Users Assigned to Database ......................... 63
12.13.6 Create Database User .................................................. 65
12.13.7 Assign User to Database ............................................... 65
12.13.8 Update Database User Privileges .................................... 66
12.13.9 Change Database User Password ................................. 67
12.13.10 Unassign User from Database ...................................... 68
12.13.11 Delete Database User .................................................. 68
12.14 Email Accounts ............................................................. 69
12.14.1 Get List of Email Accounts ............................................ 69
12.14.2 Get List of Email Accounts for Specific Domain .............. 70
12.14.3 Create Email Account .................................................. 71
12.14.4 Delete Email Account for Default Domain .................... 71
12.14.5 Delete Email Account for Specific Domain .................... 72
12.15 Services ................................................................. 73
12.15.1 Get List of Services ..................................................... 73
12.15.2 Start/Stop/Restart Services .......................................... 74
13 Assets ............................................................................. 76
13.1 Get List of Assets ............................................................ 76
13.2 Get Asset Details ............................................................ 76
13.3 Get List of Unassigned Assets .......................................... 78
14 Auto-Backups .................................................................... 83
14.1 Auto-backup Presets ....................................................... 83
14.1.1 Get List of Auto-backup Presets ........................................ 83
14.1.2 Get Auto-backup Preset Details ......................................... 84
14.1.3 Edit Auto-backup Preset ................................................ 85
14.2 Manage Auto-Backups ..................................................... 86
14.2.1 Enable Auto-backups for VS ............................................ 86
14.2.2 Disable Auto-backups for VS ........................................... 86
14.2.3 Enable Auto-backups for Disk .......................................... 87
14.2.4 Disable Auto-backups for Disk ......................................... 87
14.3 Schedules ....................................................................... 88
14.3.1 Get List of All Schedules ................................................ 88
14.3.2 Get Schedule Details ..................................................... 90
14.3.3 Get List of Schedules for a Disk ....................................... 92
14.3.4 Get List of Virtual Server Schedules ............................... 93
14.3.5 Add Schedule to Disk ..................................................... 97
14.3.6 Add Schedule to Virtual Server ....................................... 97
14.3.7 Edit Disk Schedule ........................................................ 98
14.3.8 Edit Virtual Server Schedule ......................................... 99
14.3.9 Delete Disk Schedule ..................................................... 100
14.3.10 Delete Virtual Server Schedule ...................................... 100
15 Backups/ Snapshots ........................................................ 101
15.1 Get List of All VS Backups ................................................. 101
15.2 Get List of Normal Backups .............................................. 103
15.3 Get List of Incremental Backups ....................................... 105
15.4 Get List of Disk Backups .................................................. 107
15.5 Create Incremental Backup ............................................. 109
15.6 Create Disk Backup ........................................................ 109
15.7 Create Backups for All Disks .......................................... 110
15.8 Convert Backup to Template ............................................ 111
15.9 Delete Backup .............................................................. 111
15.10 Restore Backup ............................................................ 112
15.11 Add/Edit Backup Note ..................................................... 112
16 Backup Resource Auto Backup Presets .............................. 114
16.1 Get List of Backup Resource Auto Backup Presets .......... 114
16.2 Get Backup Resource Auto Backup Preset Details ............ 115
16.3 Add Backup Resource Hourly Auto Backup Preset ............. 117
16.4 Add Backup Resource Daily Auto Backup Preset .............. 117
16.5 Add Backup Resource Weekly Auto Backup Preset .......... 118
16.6 Add Backup Resource Monthly Auto Backup Preset ................................................. 119
16.7 Add Backup Resource Yearly Auto Backup Preset .................................................. 120
16.8 Edit Backup Resource Hourly Auto Backup Preset .................................................. 120
16.9 Edit Backup Resource Daily Auto Backup Preset .................................................... 121
16.10 Edit Backup Resource Weekly Auto Backup Preset ............................................. 122
16.11 Edit Backup Resource Monthly Auto Backup Preset ............................................ 123
16.12 Add Backup Resource Yearly Auto Backup Preset ............................................. 124
16.13 Delete Backup Resource Monthly Auto Backup Preset ........................................ 124

17 Backup Resources .................................................................................................. 126
17.1 Get List of Backup Resources ............................................................................. 126
17.2 Get Backup Resource Details ........................................................................... 127
17.3 Add Backup Resource ........................................................................................ 128
17.4 Edit Backup Resource ....................................................................................... 129
17.5 Edit Backup Resource Advanced Options ....................................................... 130
17.6 Delete Backup Resource .................................................................................... 131

18 Backup Resource Zones ....................................................................................... 132
18.1 Get List of Backup Resource Zones .................................................................... 132
18.2 Get Backup Resource Zone Details ..................................................................... 133
18.3 Add Backup Resource Zone ............................................................................... 133
18.4 Edit Backup Resource Zone .............................................................................. 134
18.5 Add Backup Resource to Backup Resource Zone .............................................. 135
18.6 Remove Backup Resource from Backup Resource Zone ..................................... 135
18.7 Delete Backup Resource Zone ............................................................................ 136

19 Backup Servers ..................................................................................................... 137
19.1 Get List of Backup Servers ................................................................................ 137
19.2 Get Backup Server Details .............................................................................. 139
19.3 Get Integrated Storage Settings Details ........................................................... 141
19.4 Add Backup Server .......................................................................................... 141
19.5 Edit Backup Server .......................................................................................... 143
19.6 Edit Integrated Storage Settings ....................................................................... 143
19.7 Delete Backup Server ....................................................................................... 144
19.8 Search Backups ................................................................................................. 144
19.9 Create CloudBoot Backup Server ...................................................................... 147

20 Backup Server Zones ........................................................................................... 149
20.1 Get List of Backup Server Zones ....................................................................... 149
20.2 Get Backup Server Zone Details ....................................................................... 149
20.3 Add Backup Server Zone .................................................................................. 150
20.4 Edit Backup Server Zone .................................................................................. 151
20.5 Delete Backup Server Zone ............................................................................... 152
20.6 Get List of Servers Assigned to Backup Server Zone ....................................... 152
20.7 Assign Backup Server to Backup Server Zone .................................................. 153
20.8 Unassign Backup Server from Backup Server Zone .......................................... 154

21 Baremetal Servers ................................................................................................ 155
21.1 Get List of Baremetal Servers .......................................................................... 155
21.2 Get Baremetal Server Details ........................................................................... 158
21.3 Create Baremetal Server .................................................................................. 161
21.4 Delete Baremetal Server .................................................................................. 162
21.5 Add/Edit Admin/User Note for Baremetal Server ............................................ 162
21.6 Enable Recovery Mode for Baremetal Server .................................................. 163
21.7 Disable Recovery Mode for Baremetal Server .................................................. 164

22 Buckets .................................................................................................................. 165
22.1 Get List of Buckets ........................................................................................... 165
22.2 Get Bucket Details ......................................................................................... 166
22.3 Add Bucket ....................................................................................................... 167
22.4 Edit Bucket ....................................................................................................... 167
22.5 Delete Bucket ................................................................................................... 168
22.6 Clone Bucket ................................................................................................. 168
22.7  Access Control .................................................................................................................................. 169
22.7.1 Get List of Access Controls for Baremetal Server Type .............................................................. 169
22.7.2 Get List of Access Controls for Smart Server Type ...................................................................... 170
22.7.3 Get List of Access Controls for Virtual Server Type ...................................................................... 173
22.7.4 Get List of Access Controls for Other Server Type ....................................................................... 178
22.7.5 Add Access Control for Baremetal Server Type ........................................................................... 179
22.7.6 Add Access Control for Smart Server Type .................................................................................. 181
22.7.7 Add Access Control for Virtual Server Type .................................................................................. 183
22.7.8 Add Access Control for Other Server Type .................................................................................. 188
22.7.9 Edit Access Control for Baremetal Server Type .......................................................................... 190
22.7.10 Edit Access Control for Smart Server Type ............................................................................... 191
22.7.11 Edit Access Control for Virtual Server Type ............................................................................... 193
22.7.12 Edit Access Control for Other Server Type ............................................................................... 198
22.7.13 Delete Resource from Access Control for Baremetal Server Type ............................................. 200
22.7.14 Delete Resource from Access Control for Smart Server Type ................................................ 201
22.7.15 Delete Resource from Access Control for Virtual Server Type ................................................ 202
22.7.16 Delete Resource from Access Control for Other Server Type .................................................. 203
22.8  Rate Card .......................................................................................................................................... 204
22.8.1 Get List of Rate Cards for Baremetal Server Type ....................................................................... 204
22.8.2 Get List of Rate Cards for Smart Server Type .............................................................................. 205
22.8.3 Get List of Rate Cards for Virtual Server Type ............................................................................ 211
22.8.4 Get List of Rate Cards for Other Server Type .............................................................................. 222
22.8.5 Add Rate Cards for Baremetal Server Type ................................................................................. 225
22.8.6 Add Rate Cards for Smart Server Type ......................................................................................... 226
22.8.7 Add Rate Cards for Virtual Server Type ......................................................................................... 231
22.8.8 Add Rate Cards for Other Server Type .......................................................................................... 242
22.8.9 Edit Rate Cards for Baremetal Server Type ................................................................................... 245
22.8.10 Edit Rate Cards for Smart Server Type ....................................................................................... 249
22.8.11 Edit Rate Cards for Virtual Server Type ...................................................................................... 251
22.8.12 Edit Rate Cards for Other Server Type ........................................................................................ 262
22.8.13 Delete Resources from Rate Cards for Baremetal Server Type ................................................ 264
22.8.14 Delete Resources from Rate Cards for Smart Server Type ......................................................... 265
22.8.15 Delete Resources from Rate Cards for Virtual Server Type ....................................................... 266
22.8.16 Delete Resource from Rate Card for Other Server Type ............................................................ 267
23  Check Password Strength ...................................................................................................................... 269
24  CloudBoot IP Addresses ....................................................................................................................... 270
24.1  Get List of CloudBoot IP Addresses ............................................................................................... 270
24.2  Add CloudBoot IP Address ............................................................................................................ 271
24.3  Delete CloudBoot IP Address ......................................................................................................... 272
24.4  Create CloudBoot IP Net ............................................................................................................... 272
24.5  Create CloudBoot IP Range ............................................................................................................ 273
25  Compute Resources ............................................................................................................................... 274
25.1  Get List of Compute Resources ........................................................................................................ 274
25.2  Get Compute Resource Details ....................................................................................................... 279
25.3  Add Xen/KVM Compute Resource .................................................................................................. 284
25.4  Add Static Compute Resource ........................................................................................................ 286
25.5  Add CloudBoot Compute Resource ................................................................................................ 287
25.6  Add Smart CloudBoot Compute Resource ....................................................................................... 293
25.7  Add Baremetal CloudBoot Compute Resource ............................................................................... 298
25.8  Add VMware Compute Resource .................................................................................................. 299
25.9  Edit Xen/KVM Compute Resource .................................................................................................. 300
25.10 Edit Static Compute Resource ....................................................................................................... 301
25.11 Edit CloudBoot Compute Resource ................................................................................................ 302
25.12 Edit Smart CloudBoot Compute Resource ..................................................................................... 305
25.13 Edit Baremetal CloudBoot Compute Resource ............................................................................. 308
25.14 Edit VMware Compute Resource .................................................................................................. 309
25.15 Reboot Compute Resource ............................................................................................................. 310
25.16 Delete Compute Resource .............................................................................................................. 311
25.17 Get List of Appliances Running on Compute Resource .................................................................. 312
25.18 Get List of Data Store Joins Attached to Compute Resource ......................................................... 312
25.19 Get List of Data Stores Attached to Compute Resource ................................................................. 313
25.20 Add Data Store Join to Compute Resource ...................................................................................... 314
25.21 Remove Data Store Join from Compute Resource ........................................... 315
25.22 Get List of Compute Resource Network Joins .................................................. 315
25.23 Add Network Join to Compute Resource ......................................................... 316
25.24 Remove Network Join from Compute Resource ................................................ 317
25.25 Enable/Disable Open vSwitch ................................................................. 317
25.26 Power Cycle CloudBoot Compute Resource .................................................... 318
25.27 Get CPU Quota for Compute Resource ......................................................... 318
25.28 Edit CPU Quota for Compute Resource ......................................................... 319
25.29 Enable Kernel Crash Dumping ................................................................... 319
25.30 Enable Maintenance Mode for Xen/KVM Compute Resource ...................... 320
25.31 Disable Maintenance Mode for Xen/KVM Compute Resource ...................... 321
25.32 Add Backup Server to Compute Resource ...................................................... 321
25.33 Remove Backup Server from Compute Resource ............................................ 322
25.34 Enable/Disable Compute Zone Custom Config ............................................. 323
25.35 Power On Virtual Servers on Xen/KVM Compute Resource ....................... 323
25.36 Power Off Virtual Servers on Xen/KVM Compute Resource ....................... 324
25.37 Enable Storage Related Services for CloudBoot Compute Resources .......... 325
25.38 Disable Storage Related Services for CloudBoot Compute Resources .......... 326
25.39 Edit Static Compute Resource Devices ......................................................... 326
25.40 Get Details of Integrated Storage Settings .................................................... 327
25.41 Edit Integrated Storage Settings on Compute Resource ............................. 328

26 Compute Zones .......................................................................................... 330
 26.1 Get List of Compute Zones .......................................................................... 330
 26.2 Get Compute Zone Details ............................................................................ 333
 26.3 Add Compute Zone ....................................................................................... 336
 26.3.1 New API Requests ...................................................................................... 337
 26.4 Edit Compute Zone ....................................................................................... 340
 26.4.1 New API Requests ...................................................................................... 341
 26.5 Delete Compute Zone .................................................................................... 343
 26.6 Get List of Compute Resources Attached to Compute Zone ...................... 343
 26.7 Attach Compute Resource to Compute Zone ................................................ 344
 26.8 Remove Compute Resource from Compute Zone ......................................... 345
 26.9 Get List of Data Store Joins Attached to Compute Zone .............................. 345
 26.10 Add Data Store Join to Compute Zone .......................................................... 346
 26.11 Remove Data Store Join from Compute Zone ............................................. 346
 26.12 Get List of Network Joins Attached to Compute Zone .................................. 347
 26.13 Add Network Join to Compute Zone ............................................................. 347
 26.14 Remove Network Join from Compute Zone .................................................. 348
 26.15 Update CPU Flags for Compute Zone .......................................................... 349
 26.16 Get Extended CPU Configuration Details for Compute Zone .................. 349
 26.17 Add Backup Server to Compute Zone .......................................................... 350
 26.18 Remove Backup Server from Compute Zone ............................................. 350
 26.19 Edit Compute Zone Custom Config ............................................................. 351
 26.20 Get List of Backup Resource Zones Attached to Compute Zone ............. 351
 26.21 Add Backup Resource Zone to Compute Zone ........................................... 352
 26.22 Remove Backup Resource Zone from Compute Zone .................................. 353
 26.23 Enable/Disable Failover ............................................................................. 354

27 Container Servers ....................................................................................... 357
 27.1 Get List of All Container Servers ................................................................. 357
 27.2 Get Container Server Details ....................................................................... 362
 27.3 Get Statuses for all Container Servers .......................................................... 367
 27.4 Get Container Server Status ....................................................................... 368
 27.5 Get Container Server Cloud Config ............................................................. 369
 27.6 Add Container Server .................................................................................... 370
 27.7 Add Container Server Cloud Config ............................................................. 370
 27.8 View Encrypted Container Server Password .............................................. 372
 27.9 Build or Rebuild Container Server ............................................................... 373
 27.10 Edit Container Server .................................................................................. 374
 27.11 Edit Container Server Cloud Config ............................................................ 376
27.12 Change Container Server Owner ................................................................. 376
27.13 Reset Container Server Root Password ...................................................... 377
27.14 Migrate Container Server ........................................................................... 378
27.15 Set VIP Status for Container Server ............................................................ 379
27.16 Delete Container Server ............................................................................. 379
27.17 Start up Container Server ........................................................................... 380
27.18 Segregate Container Server ........................................................................ 381
27.19 Desegregate Container Server ..................................................................... 381
27.20 Reboot Container Server ............................................................................ 382
27.21 Reboot Container Server in Recovery .......................................................... 382
27.22 Boot Container Server from ISO .................................................................. 383
27.23 Suspend Container Server ........................................................................... 383
27.24 Unlock Container Server ............................................................................ 383
27.25 Unsuspend Container Server ....................................................................... 384
27.26 Shut down Container Server ....................................................................... 384
27.27 Stop Container Server ................................................................................. 384
27.28 Open Container Server Console ................................................................ 385
27.29 Container Server Billing Statistics ............................................................... 385
27.30 Search Container Server by Label ............................................................... 388
27.31 Get Container Server CPU Usage Statistics .......... ................................. 388
27.32 Add/Edit Admin/User Note for Container Server ........................................ 389
27.32.1 Add/Edit User Note ................................................................................. 390

28 Control Panel Maintenance .......................................................................... 392
28.1 Get Control Panel Maintenance Status ........................................................ 392
28.2 Enable Control Panel Maintenance .............................................................. 392
28.3 Disable Control Panel Maintenance ............................................................. 393

29 Currencies .................................................................................................... 394
29.1 Get List of Currencies.................................................................................. 394
29.2 Get Currency Details ................................................................................... 395
29.3 Add Currency .............................................................................................. 396
29.4 Edit Currency .............................................................................................. 397
29.5 Delete Currency ........................................................................................... 397

30 Custom Recipe Variables .............................................................................. 399
30.1 Get List of Custom Variables ....................................................................... 399
30.2 Get Custom Variable Details ...................................................................... 400
30.3 Edit Custom Variable .................................................................................. 401
30.4 Add Custom Variable .................................................................................. 401
30.5 Delete Custom Variable .............................................................................. 402
30.6 Get List of Virtual Server Custom Variables .............................................. 403
30.7 Get List of Smart Server Custom Variables ................................................. 404
30.8 Get List of Baremetal Server Variables ....................................................... 405
30.9 Get Virtual Server Custom Variable Details ............................................... 406
30.10 Get Smart Server Custom Variable Details ............................................... 407
30.11 Get Baremetal Server Custom Variable Details .......................................... 408
30.12 Add Virtual Server Custom Variable ......................................................... 409
30.13 Add Smart Server Custom Variable ............................................................ 410
30.14 Add Baremetal Server Custom Variable .................................................... 411
30.15 Edit Virtual Server Custom Variable ......................................................... 411
30.16 Edit Smart Server Custom Variable ........................................................... 412
30.17 Edit Baremetal Server Custom Variable .................................................... 413
30.18 Delete Virtual Server Custom Variable ...................................................... 414
30.19 Delete Smart Server Custom Variable ....................................................... 414
30.20 Delete Baremetal Server Custom Variable ................................................ 415

31 Data Stores .................................................................................................. 416
31.1 Get List of Data Stores ................................................................................ 416
31.2 Get Data Store Details ................................................................................ 417
31.3 Add LVM Data Store ................................................................................... 418
31.4 Add VMware Data Store ............................................................................. 419
### 32 Data Store Zones

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>32.1</td>
<td>Get List of Data Store Zones</td>
<td>425</td>
</tr>
<tr>
<td>32.2</td>
<td>Get Data Store Zone Details</td>
<td>426</td>
</tr>
<tr>
<td>32.3</td>
<td>Add Data Store Zone</td>
<td>427</td>
</tr>
<tr>
<td>32.4</td>
<td>Edit Data Store Zone</td>
<td>428</td>
</tr>
<tr>
<td>32.5</td>
<td>Delete Data Store Zone</td>
<td>429</td>
</tr>
<tr>
<td>32.6</td>
<td>Get the List of Data Stores Attached to Data Store Zone</td>
<td>430</td>
</tr>
<tr>
<td>32.7</td>
<td>Attach Data Store to Data Store Zone</td>
<td>430</td>
</tr>
<tr>
<td>32.8</td>
<td>Detach Data Store from Data Store Zone</td>
<td>431</td>
</tr>
</tbody>
</table>

### 33 Disks

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>33.1</td>
<td>Get List of Disks</td>
<td>432</td>
</tr>
<tr>
<td>33.2</td>
<td>Get List of VS Disks</td>
<td>433</td>
</tr>
<tr>
<td>33.3</td>
<td>Get VS Disk Details</td>
<td>435</td>
</tr>
<tr>
<td>33.4</td>
<td>Add New Disk</td>
<td>437</td>
</tr>
<tr>
<td>33.5</td>
<td>Edit Disk</td>
<td>438</td>
</tr>
<tr>
<td>33.6</td>
<td>Edit Disk IOPS Limits</td>
<td>440</td>
</tr>
<tr>
<td>33.7</td>
<td>Migrate Disks</td>
<td>440</td>
</tr>
<tr>
<td>33.8</td>
<td>Delete Disk</td>
<td>441</td>
</tr>
<tr>
<td>33.9</td>
<td>View Disk IOPS</td>
<td>442</td>
</tr>
<tr>
<td>33.10</td>
<td>Build Disk</td>
<td>443</td>
</tr>
<tr>
<td>33.11</td>
<td>Unlock Disk</td>
<td>444</td>
</tr>
<tr>
<td>33.12</td>
<td>Get List of Backups Available for Disk</td>
<td>444</td>
</tr>
<tr>
<td>33.13</td>
<td>Assign Disk to VS</td>
<td>446</td>
</tr>
<tr>
<td>33.14</td>
<td>Unassign Disk from VS</td>
<td>446</td>
</tr>
</tbody>
</table>

### 34 DRaaS Dashboard

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>34.1</td>
<td>API Credentials (DRaaS)</td>
<td>447</td>
</tr>
<tr>
<td>34.1.1</td>
<td>Get Cloud API Credentials (DRaaS)</td>
<td>447</td>
</tr>
<tr>
<td>34.1.2</td>
<td>Update Cloud API Credentials (DRaaS)</td>
<td>447</td>
</tr>
<tr>
<td>34.2</td>
<td>Cloud (DRaaS)</td>
<td>448</td>
</tr>
<tr>
<td>34.2.1</td>
<td>Register New Cloud (DRaaS)</td>
<td>448</td>
</tr>
<tr>
<td>34.2.2</td>
<td>Get List of Clouds (DRaaS)</td>
<td>449</td>
</tr>
<tr>
<td>34.2.3</td>
<td>Get Cloud Details (DRaaS)</td>
<td>450</td>
</tr>
<tr>
<td>34.2.4</td>
<td>Update Cloud (DRaaS)</td>
<td>450</td>
</tr>
<tr>
<td>34.2.5</td>
<td>Remove Cloud from Dashboard (DRaaS)</td>
<td>451</td>
</tr>
<tr>
<td>34.3</td>
<td>Compute Zones (DRaaS)</td>
<td>451</td>
</tr>
<tr>
<td>34.3.1</td>
<td>Get List of Compute Zones (DRaaS)</td>
<td>452</td>
</tr>
<tr>
<td>34.3.2</td>
<td>Get Compute Zone Details (DRaaS)</td>
<td>453</td>
</tr>
<tr>
<td>34.3.3</td>
<td>Register Compute Zone (DRaaS)</td>
<td>453</td>
</tr>
<tr>
<td>34.3.4</td>
<td>Update Compute Zone (DRaaS)</td>
<td>454</td>
</tr>
<tr>
<td>34.3.5</td>
<td>Delete Compute Zone (DRaaS)</td>
<td>455</td>
</tr>
<tr>
<td>34.3.6</td>
<td>Get List of Compute Zone Links (DRaaS)</td>
<td>455</td>
</tr>
<tr>
<td>34.3.7</td>
<td>Get Compute Zone Link Details (DRaaS)</td>
<td>456</td>
</tr>
<tr>
<td>34.3.8</td>
<td>Create Compute Zone Link (DRaaS)</td>
<td>456</td>
</tr>
<tr>
<td>34.3.9</td>
<td>Get Compute Zone Private Key (DRaaS)</td>
<td>457</td>
</tr>
<tr>
<td>34.3.10</td>
<td>Regenerate Compute Zone Private Key (DRaaS)</td>
<td>457</td>
</tr>
<tr>
<td>34.3.11</td>
<td>Delete Compute Zone Link (DRaaS)</td>
<td>458</td>
</tr>
<tr>
<td>34.4</td>
<td>Disks (DRaaS)</td>
<td>458</td>
</tr>
<tr>
<td>34.4.1</td>
<td>Get List of Disks (DRaaS)</td>
<td>458</td>
</tr>
<tr>
<td>34.4.2</td>
<td>Get Disk Details (DRaaS)</td>
<td>459</td>
</tr>
<tr>
<td>34.5</td>
<td>Events (DRaaS)</td>
<td>460</td>
</tr>
<tr>
<td>34.5.1</td>
<td>Get List of Events (DRaaS)</td>
<td>460</td>
</tr>
<tr>
<td>34.5.2</td>
<td>Get Event Details (DRaaS)</td>
<td>461</td>
</tr>
<tr>
<td>34.6</td>
<td>Get DRaaS Dashboard Version</td>
<td>462</td>
</tr>
<tr>
<td>34.7</td>
<td>IP Ranges (DRaaS)</td>
<td>462</td>
</tr>
<tr>
<td>34.7.1</td>
<td>Get List of Cloud IP Ranges</td>
<td>462</td>
</tr>
<tr>
<td>34.7.2</td>
<td>Get Cloud IP Range Details</td>
<td>463</td>
</tr>
</tbody>
</table>
34.7.3 Create Cloud IP Range (DRaaS) ................................................................. 463
34.7.4 Update Cloud IP Range (DRaaS) ............................................................ 464
34.7.5 Delete Cloud IP Range (DRaaS) ............................................................. 464
34.8 Locations (DRaaS) ..................................................................................... 465
34.8.1 Get Locations List .................................................................................. 465
34.8.2 Get Location Details .............................................................................. 466
34.9 Networks (DRaaS) ..................................................................................... 466
34.9.1 Get List of Networks (DRaaS) ............................................................... 466
34.9.2 Get Network Details (DRaaS) ............................................................... 467
34.9.3 Update Network (DRaaS) ...................................................................... 468
34.9.4 Get List of IP Addresses (DRaaS) ......................................................... 469
34.9.5 Get IP Address Details (DRaaS) ............................................................ 469
34.9.6 Get Network Interfaces List (DRaaS) ................................................... 470
34.9.7 Get Network Interface Details (DRaaS) ................................................ 471
34.9.8 Network Links (DRaaS) ......................................................................... 471
34.10 References (DRaaS) ................................................................................. 474
34.10.1 Get List of References .......................................................................... 474
34.10.2 Get Reference Details ........................................................................... 476
34.11 Users (DRaaS) .......................................................................................... 477
34.11.1 Get List of Users (DRaaS) ..................................................................... 477
34.11.2 Get User Details (DRaaS) .................................................................... 478
34.11.3 Register New User ............................................................................... 478
34.11.4 Update User (DRaaS) .......................................................................... 479
34.11.5 Delete User (DRaaS) ........................................................................... 480
34.12 Virtual Machines (DRaaS) ....................................................................... 480
34.12.1 Get List of Virtual Machines (DRaaS) .................................................. 480
34.12.2 Get Virtual Machine Details (DRaaS) ................................................... 483
34.12.3 Register Virtual Machine (DRaaS) ....................................................... 484
34.12.4 Perform Virtual Machine Failover (DRaaS) ......................................... 485
34.12.5 Start Virtual Machine Failback (DRaaS) .............................................. 485
34.12.6 Finalize Virtual Machine Failback (DRaaS) ........................................ 486
34.12.7 Cancel Virtual Machine Failback (DRaaS) ........................................... 486
34.12.8 Remove Virtual Machine (DRaaS) ....................................................... 486

35 Embed Statistics Charts ............................................................................. 488

36 Errors .......................................................................................................... 491
36.1 Get List of Errors ....................................................................................... 491
36.2 Get Error Details ....................................................................................... 493

37 Federation ..................................................................................................... 495
37.1 Add Zone to Federation ............................................................................ 495
37.2 Enable Federated Zone .............................................................................. 497
37.3 Disable Federated Zone ............................................................................ 497
37.4 Remove Zone from Federation ................................................................. 498
37.5 Get List of Federated Resources ............................................................... 498
37.6 Get Federated Resource Details ............................................................... 501
37.7 Subscribe to Federated Zone ..................................................................... 502
37.8 Unsubscribe from Federated Zone ........................................................... 503
37.9 Suspend Zone ......................................................................................... 503
37.10 Unsuspend Zone .................................................................................... 504

38 Firewall Rules for VSs ................................................................................ 505
38.1 Get List of Firewall Rules ......................................................................... 505
38.2 Apply Firewall Rule .................................................................................. 506
38.3 Add Firewall Rule .................................................................................... 506
38.4 Edit Firewall Rule ..................................................................................... 509
38.5 Delete Firewall Rule ................................................................................ 511
38.6 Change Firewall Rule Position .................................................................. 512
38.7 Set Default Firewall Rules ....................................................................... 513

39 Firewalls ....................................................................................................... 515
39.1 Get List of Firewalls ................................................................................ 515
39.2 Get Firewall Details ................................................................................. 516
39.3 Add Firewall ............................................................................................. 517
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>44</td>
<td>IP Addresses</td>
<td>567</td>
</tr>
<tr>
<td>44.1</td>
<td>Get List of IP Addresses in IP Range</td>
<td>567</td>
</tr>
<tr>
<td>44.2</td>
<td>Assign IP Address to User</td>
<td>568</td>
</tr>
<tr>
<td>44.3</td>
<td>Unassign IP Address from User</td>
<td>569</td>
</tr>
<tr>
<td>44.4</td>
<td>Get List of IP Addresses for VS</td>
<td>570</td>
</tr>
<tr>
<td>44.5</td>
<td>Assign IP Address to VS</td>
<td>572</td>
</tr>
<tr>
<td>44.6</td>
<td>Unassign IP Address from VS</td>
<td>573</td>
</tr>
<tr>
<td>44.7</td>
<td>External IP Addresses</td>
<td>574</td>
</tr>
<tr>
<td>44.7.1</td>
<td>Get External IP Address Details</td>
<td>574</td>
</tr>
<tr>
<td>44.7.2</td>
<td>Add/Edit External IP Address</td>
<td>574</td>
</tr>
<tr>
<td>44.7.3</td>
<td>Delete External IP Address</td>
<td>575</td>
</tr>
<tr>
<td>44.8</td>
<td>Get IP Usage Report</td>
<td>575</td>
</tr>
<tr>
<td>45</td>
<td>IP Nets</td>
<td>577</td>
</tr>
<tr>
<td>45.1</td>
<td>Get List of IP Nets in Network</td>
<td>577</td>
</tr>
<tr>
<td>45.2</td>
<td>Create IP Net</td>
<td>578</td>
</tr>
<tr>
<td>45.3</td>
<td>Edit IP Net</td>
<td>579</td>
</tr>
<tr>
<td>45.4</td>
<td>Delete IP Net</td>
<td>579</td>
</tr>
<tr>
<td>46</td>
<td>IP Ranges</td>
<td>580</td>
</tr>
<tr>
<td>46.1</td>
<td>Get List of IP Ranges in IP Net</td>
<td>580</td>
</tr>
<tr>
<td>46.2</td>
<td>Create IP Range</td>
<td>581</td>
</tr>
<tr>
<td>46.3</td>
<td>Edit IP Range</td>
<td>581</td>
</tr>
<tr>
<td>46.4</td>
<td>Delete IP Range</td>
<td>582</td>
</tr>
<tr>
<td>47</td>
<td>ISOs</td>
<td>583</td>
</tr>
<tr>
<td>47.1</td>
<td>Get List of ISOs</td>
<td>583</td>
</tr>
<tr>
<td>47.2</td>
<td>Get ISO Details</td>
<td>585</td>
</tr>
<tr>
<td>47.3</td>
<td>Get List of Public ISOs</td>
<td>587</td>
</tr>
<tr>
<td>47.4</td>
<td>Get List of ISOs of Particular User</td>
<td>589</td>
</tr>
<tr>
<td>47.5</td>
<td>Get List of User ISOs</td>
<td>591</td>
</tr>
<tr>
<td>47.6</td>
<td>Get List of Own ISOs</td>
<td>593</td>
</tr>
<tr>
<td>47.7</td>
<td>Update ISO</td>
<td>595</td>
</tr>
<tr>
<td>47.8</td>
<td>Add New ISO</td>
<td>596</td>
</tr>
<tr>
<td>47.9</td>
<td>Make ISO Public</td>
<td>597</td>
</tr>
<tr>
<td>47.10</td>
<td>Delete ISO</td>
<td>598</td>
</tr>
<tr>
<td>48</td>
<td>License</td>
<td>599</td>
</tr>
<tr>
<td>48.1</td>
<td>Get License Details</td>
<td>599</td>
</tr>
<tr>
<td>48.2</td>
<td>Edit License Details</td>
<td>600</td>
</tr>
<tr>
<td>49</td>
<td>Load Balancers</td>
<td>602</td>
</tr>
<tr>
<td>49.1</td>
<td>Get List of Load Balancers</td>
<td>602</td>
</tr>
<tr>
<td>49.2</td>
<td>Get Load Balancer Details</td>
<td>606</td>
</tr>
<tr>
<td>49.3</td>
<td>Get the List of Load Balancing Clusters</td>
<td>609</td>
</tr>
<tr>
<td>49.4</td>
<td>Get Load Balancing Cluster Details</td>
<td>617</td>
</tr>
<tr>
<td>49.5</td>
<td>Get Load Balancer Billing Statistics</td>
<td>624</td>
</tr>
<tr>
<td>49.6</td>
<td>Get List of Load Balancer Autoscaling Monitors</td>
<td>628</td>
</tr>
<tr>
<td>49.7</td>
<td>Get Load Balancer Autoscaling Monitor Details</td>
<td>632</td>
</tr>
<tr>
<td>49.8</td>
<td>Add Load Balancing Cluster</td>
<td>636</td>
</tr>
<tr>
<td>49.9</td>
<td>Add Autoscaling Cluster</td>
<td>637</td>
</tr>
<tr>
<td>49.10</td>
<td>Add Nodes to Cluster Type</td>
<td>639</td>
</tr>
<tr>
<td>49.11</td>
<td>Remove Nodes from Cluster Type</td>
<td>640</td>
</tr>
<tr>
<td>49.12</td>
<td>Edit Load Balancing Cluster</td>
<td>641</td>
</tr>
<tr>
<td>49.13</td>
<td>Edit Autoscaling Cluster</td>
<td>641</td>
</tr>
<tr>
<td>49.14</td>
<td>Edit Load Balancing Cluster Ports</td>
<td>643</td>
</tr>
<tr>
<td>49.15</td>
<td>Delete Load Balancing Cluster</td>
<td>644</td>
</tr>
<tr>
<td>49.16</td>
<td>Rebuild Load Balancer</td>
<td>644</td>
</tr>
<tr>
<td>49.17</td>
<td>Search Load Balancer by Label</td>
<td>645</td>
</tr>
<tr>
<td>49.18</td>
<td>Start up Load Balancer</td>
<td>645</td>
</tr>
<tr>
<td>49.19</td>
<td>Shut Down Load Balancer</td>
<td>645</td>
</tr>
<tr>
<td>49.20</td>
<td>Suspend Load Balancer</td>
<td>646</td>
</tr>
<tr>
<td>49.21</td>
<td>Stop Load Balancer</td>
<td>646</td>
</tr>
</tbody>
</table>
50 Locales ................................................................................................................. 648
51 Location Groups ............................................................................................... 649
51.1 Get List of Location Groups ............................................................................ 649
51.2 Get Location Group Details ........................................................................... 650
51.3 Refresh Location Groups ................................................................................ 651
51.4 Attach Compute Zone to Location Group ...................................................... 652
51.5 Detach Compute Zone from Location Group ............................................... 652
51.6 Attach Data Store Zone to Location Group .................................................. 653
51.7 Detach Data Store Zone from Location Group ............................................. 653
51.8 Attach Network Zone to Location Group ...................................................... 654
51.9 Detach Network Zone from Location Group ................................................ 654
51.10 Attach Backup Server Zone to Location Group ........................................... 655
51.11 Detach Backup Server Zone from Location Group ....................................... 655
51.12 Get List of Compute Zones Attached to Location Group ......................... 656
51.13 Get List of Data Store Zones Attached to Location Group ....................... 659
51.14 Get List of Network Zones Attached to Location Group ......................... 660
51.15 Get List of Backup Server Zones Attached to Location Group ................. 661
52 Logs ..................................................................................................................... 662
52.1 Get List of Log Items ....................................................................................... 662
52.2 Get Log Item Details ...................................................................................... 663
52.3 Get List of VS Log Items ................................................................................ 664
52.4 Get VS Log Item Details ................................................................................. 665
52.5 Get List of Resource Differences ................................................................... 666
52.6 Get Resource Difference Details .................................................................. 667
53 Look&Feel .......................................................................................................... 670
53.1 Look&Feel Themes ......................................................................................... 670
53.1.1 Get List of Look&Feel Themes ................................................................. 670
53.1.2 Get Look&Feel Theme Details .................................................................. 672
53.1.3 Add Look&Feel Theme ............................................................................ 674
53.1.4 Edit Look&Feel Theme ............................................................................. 675
53.1.5 Delete Look&Feel Theme ......................................................................... 675
53.2 Service Insertion Groups ................................................................................ 675
53.2.1 Get List of Service Insertion Groups ....................................................... 676
53.2.2 Get Service Insertion Group Details ....................................................... 676
53.2.3 Add Service Insertion Group ................................................................... 677
53.2.4 Edit Service Insertion Group ................................................................... 678
53.2.5 Delete Service Insertion Group ................................................................ 678
53.3 Service Insertion Pages ................................................................................ 678
53.3.1 Get List of Service Insertion Pages .......................................................... 679
53.3.2 Get Service Insertion Page Details .......................................................... 680
53.3.3 Add Service Insertion Page ...................................................................... 681
53.3.4 Edit Service Insertion Page ...................................................................... 682
53.3.5 Delete Service Insertion Page .................................................................. 682
54 My Template Groups ......................................................................................... 684
54.1 Get My Template Groups List ....................................................................... 684
54.2 Get My Template Group Details ................................................................... 687
54.3 Add My Template Group ................................................................................ 688
54.4 Add Child Template Group ........................................................................... 690
54.5 Edit My Template Group ............................................................................... 690
54.6 Delete My Template Group ........................................................................... 691
54.7 Get List of Templates Attached to Template Group ..................................... 691
54.8 Attach Template to Template Group ............................................................. 694
54.9 Detach Template from Template Group ....................................................... 695
55 Network Interfaces ............................................................................................. 696
55.1 Get List of VS Network Interfaces .................................................................. 696
55.2 Get Network Interface Details ....................................................................... 697
55.3 Add Network Interface to VS ........................................................................ 697
55.4 Edit Network Interface .................................................................................. 698
| 55.5 | Delete Network Interface ................................................. | 698 |
| 55.6 | Get VS Network Interface Usage Statistics ........................ | 699 |
| 56 | **Networks** ....................................................................... | 701 |
| 56.1 | Get List of Networks ........................................................ | 701 |
| 56.2 | Get Network Details .......................................................... | 702 |
| 56.3 | Add Network .................................................................. | 702 |
| 56.4 | Edit Network ................................................................. | 703 |
| 56.5 | Rebuild VS Network .......................................................... | 704 |
| 56.6 | Delete Network ................................................................. | 704 |
| 57 | **Network Zones** ............................................................. | 706 |
| 57.1 | Get List of Network Zones .................................................. | 706 |
| 57.2 | Get Network Zone Details ................................................... | 707 |
| 57.3 | Add Network Zone ............................................................. | 708 |
| 57.4 | Edit Network Zone ............................................................ | 709 |
| 57.5 | Delete Network Zone .......................................................... | 709 |
| 57.6 | Attach Network to Network Zone .......................................... | 710 |
| 57.7 | Remove Network from Network Zone ...................................... | 710 |
| 58 | **Notifications** ............................................................... | 712 |
| 58.1 | Event Types .................................................................. | 712 |
| 58.1.1 | Get List of Events .......................................................... | 712 |
| 58.1.2 | Get List of Event Types ................................................... | 713 |
| 58.1.3 | Add Custom Event Type ................................................... | 714 |
| 58.1.4 | Edit Custom Event Type ................................................... | 715 |
| 58.1.5 | Trigger Custom Event Type ................................................ | 716 |
| 58.1.6 | Delete Custom Event Type ................................................ | 716 |
| 58.2 | External Recipients ........................................................... | 716 |
| 58.2.1 | Get List of External Recipients ........................................... | 716 |
| 58.2.2 | Add External Recipient ..................................................... | 717 |
| 58.2.3 | Edit External Recipient ..................................................... | 718 |
| 58.2.4 | Delete External Recipients ................................................ | 718 |
| 58.3 | Recipients List ............................................................... | 719 |
| 58.3.1 | Get List of Recipients Lists ............................................... | 719 |
| 58.3.2 | Add Recipients List ........................................................ | 720 |
| 58.3.3 | Edit Recipients List ........................................................ | 720 |
| 58.3.4 | Delete Recipients List ....................................................... | 721 |
| 58.4 | Notification Templates ...................................................... | 721 |
| 58.4.1 | Get List of Notification Templates ...................................... | 721 |
| 58.4.2 | Add Notification Template ................................................ | 722 |
| 58.4.3 | Edit Notification Template ................................................ | 723 |
| 58.4.4 | Restore Notification Template to Default ............................ | 723 |
| 58.4.5 | Delete Notification Template ............................................ | 723 |
| 58.5 | Subscriptions ................................................................. | 724 |
| 58.5.1 | Get List of Subscriptions .................................................. | 724 |
| 58.5.2 | Add Subscription ............................................................ | 725 |
| 58.5.3 | Edit Subscription ............................................................ | 726 |
| 58.5.4 | Delete Subscription .......................................................... | 726 |
| 58.6 | Gateways ...................................................................... | 726 |
| 58.6.1 | Get List of Gateways ........................................................ | 727 |
| 58.6.2 | Add Internal Gateway ....................................................... | 728 |
| 58.6.3 | Add Sendmail Gateway ..................................................... | 729 |
| 58.6.4 | Add SMTP Gateway .......................................................... | 729 |
| 58.6.5 | Delete Gateway .............................................................. | 730 |
| 58.7 | Get List of Deliveries ........................................................ | 731 |
| 58.8 | Configuration ................................................................. | 732 |
| 58.8.1 | Enable/Disable Notifications .............................................. | 732 |
| 58.8.2 | Set Number of Notifications to Show .................................... | 732 |
| 58.8.3 | Set Notification Prefix .................................................... | 733 |
| 59 | **OnApp Engine** .............................................................. | 734 |
| 59.1 | Get OnApp Engine Status .................................................. | 734 |
| 59.2 | Start OnApp Engine ........................................................ | 735 |
| 59.3 | Stop OnApp Engine .......................................................... | 735 |
| 59.4 | Reload OnApp Engine | 736 |
| 60 | **OVAs** | 738 |
| 60.1 | Get List of OVAs | 738 |
| 60.2 | Get List of System OVAs | 741 |
| 60.3 | Get List of Own OVAs | 744 |
| 60.4 | Get List of User OVAs | 747 |
| 60.5 | Get OVA Details | 750 |
| 60.6 | Get OVA Disks | 753 |
| 60.7 | Upload OVA | 754 |
| 60.8 | Convert OVA | 755 |
| 60.9 | Unlock OVA | 756 |
| 60.10 | Edit OVA | 757 |
| 60.11 | Make OVA Public | 758 |
| 60.12 | Search OVA | 758 |
| 60.13 | Delete OVA Files | 761 |
| 60.14 | Delete OVA | 761 |
| 61 | **Pagination** | 763 |
| 62 | **Payments** | 766 |
| 62.1 | Get List of All User Payments | 766 |
| 62.2 | Get List of Payments of Particular User | 767 |
| 62.3 | Create User Payment | 768 |
| 62.4 | Edit User Payment | 768 |
| 62.5 | Delete User Payment | 769 |
| 63 | **Publishing Rules** | 770 |
| 63.1 | Get List of Publishing Rules | 770 |
| 63.2 | Add Publishing Rule | 771 |
| 63.3 | Delete Publishing Rule | 771 |
| 64 | **Recipe Groups** | 773 |
| 64.1 | Get List of Recipe Groups | 773 |
| 64.2 | Get Recipe Group Details | 775 |
| 64.3 | Add Recipe Group | 776 |
| 64.4 | Add Child Recipe Group | 776 |
| 64.5 | Edit Recipe Group | 777 |
| 64.6 | Delete Recipe Group | 777 |
| 64.7 | Get List of Recipes Attached to Recipe Group | 778 |
| 64.8 | Attach Recipe to Recipe Group | 778 |
| 64.9 | Remove Recipe from Recipe Group | 779 |
| 65 | **Recipes** | 780 |
| 65.1 | Get List of Recipes | 780 |
| 65.2 | Get List of Servers Using Recipe | 783 |
| 65.3 | Get Recipe Details | 784 |
| 65.4 | Add Recipe | 787 |
| 65.5 | Edit Recipe | 787 |
| 65.6 | Delete Recipe | 788 |
| 65.7 | Manage Recipe Steps | 788 |
| 65.7.1 | Get the List of Recipe Steps | 788 |
| 65.7.2 | Create Recipe Step | 793 |
| 65.7.3 | Edit Recipe Step | 794 |
| 65.7.4 | Remove Recipe Step | 795 |
| 65.7.5 | Swap Recipe Steps Locations | 795 |
| 65.8 | Manage Virtual Server Recipes | 795 |
| 65.8.1 | Get List of Virtual Server Recipes | 796 |
| 65.8.2 | Assign Recipe to Virtual Server | 796 |
| 65.8.3 | Remove Recipe from Virtual Server | 797 |
| 65.8.4 | Run Recipe on Multiple Virtual Servers | 798 |
| 65.9 | Manage Smart Server Recipes | 799 |
| 65.9.1 | Get List of Smart Server Recipes | 799 |
| 65.9.2 | Assign Recipe to Smart Server | 799 |
65.9.3 Remove Recipe from Smart Server ................................................................. 801
65.9.4 Run Recipe on Multiple Smart Servers ......................................................... 801
65.10 Manage Baremetal Server Recipes ............................................................... 802
65.10.1 Get the List of Baremetal Server Recipes ................................................ 802
65.10.2 Assign Recipe to Baremetal Server ......................................................... 802
65.10.3 Remove Recipe from Baremetal Server .................................................... 802
65.11 Manage Template Recipes ........................................................................... 803
65.11.1 Get the List of Template Recipes ............................................................... 803
65.11.2 Assign Recipe to Template ..................................................................... 803
65.11.3 Remove Recipe from Template ............................................................... 803
65.12 Manage Compute Zone Recipes ................................................................. 805
65.12.1 Get the List of Compute Zone Recipes .................................................. 805
65.12.2 Assign Recipe to Compute Zone ............................................................. 806
65.12.3 Remove Recipe from Compute Zone ...................................................... 807
65.13 Manage Control Panel Recipes ..................................................................... 808
65.13.1 Get the List of Control Panel Recipes ..................................................... 808
65.13.2 Assign Recipe to Control Panel ............................................................... 808
65.13.3 Remove Recipe from Control Panel .......................................................... 810

66 Resolvers ........................................................................................................... 811
66.1 Get List of Resolvers .................................................................................... 811
66.2 Get Resolver Details .................................................................................... 811
66.3 Add Resolver ............................................................................................... 812
66.4 Edit Resolver ............................................................................................... 813
66.5 Delete Resolver ........................................................................................... 813

67 Restrictions Sets ............................................................................................. 814
67.1 Get List of Restrictions Sets ......................................................................... 814
67.2 Get Restrictions Set Details ......................................................................... 816
67.3 Get List of All Restrictions Resources ....................................................... 817
67.4 Create Restrictions Set ............................................................................... 818
67.5 Edit Restrictions Set ................................................................................... 819

68 Roles .................................................................................................................. 821
68.1 Get List of Roles .......................................................................................... 821
68.2 Get Role Details ......................................................................................... 822
68.3 Add Role ....................................................................................................... 823
68.4 Edit Role ....................................................................................................... 823
68.5 Delete Role .................................................................................................... 824
68.6 Edit User Role Assignment .......................................................................... 824
68.7 Get List of All Permissions .......................................................................... 825
68.8 Clone Role ..................................................................................................... 825

69 SAML ID Providers ............................................................................................ 827
69.1 Get List of SAML ID Providers ..................................................................... 827
69.2 Get SAML ID Provider Details ..................................................................... 829

70 SDN Management ............................................................................................. 832
70.1 SDN Manager ............................................................................................... 832
70.1.1 Get SDN Manager Details ....................................................................... 832
70.1.2 Add SDN Manager .................................................................................. 833
70.1.3 Edit SDN Manager ................................................................................ 833
70.1.4 Delete SDN Manager ............................................................................. 834
70.1.5 Add Connection Option to SDN Manager ............................................. 834
70.1.6 Delete Connection Option from SDN Manager ...................................... 835
70.2 SDN Nodes .................................................................................................... 835
70.2.1 Get List of SDN Manager Nodes .............................................................. 835
70.2.2 Add Nodes to SDN Manager .................................................................. 836
70.2.3 Reattach SDN Node ............................................................................... 837
70.2.4 Delete Nodes from SDN Manager .......................................................... 837
70.3 SDN Networks ............................................................................................... 838
70.3.1 Get List of SDN Networks ....................................................................... 838
70.3.2 Get SDN Network Details ....................................................................... 839
70.3.3 Add SDN Network .................................................................................. 840
70.3.4 Connect SDN Network to SDN Node ...................................................... 840
70.3.5 Assign SDN Network to User ......................................................... 841
70.3.6 Unassign SDN Network from User .................................................. 841
70.3.7 Delete SDN Network ................................................................. 842
70.3.8 Recreate Bridges ..................................................................... 842
70.3.9 Delete Bridge ........................................................................... 843
70.3.10 Cleanup Zombie Tunnels ......................................................... 843

71 Smart Servers ................................................................................. 845
71.1 Get List of Smart Servers ............................................................... 845
71.2 Get Smart Server Details ............................................................... 848
71.3 Add Smart Server ........................................................................ 849
71.4 View Encrypted Smart Server Password ....................................... 851
71.5 Build Smart Server ..................................................................... 851
71.6 Edit Smart Server ........................................................................ 852
71.7 Change Smart Server Owner ......................................................... 853
71.8 Migrate Smart Server ................................................................... 854
71.9 Delete Smart Server ..................................................................... 855
71.10 Start up Smart Server .................................................................. 855
71.11 Reboot Smart Server ................................................................... 856
71.12 Reboot Smart Server in Recovery ............................................... 856
71.13 Suspend Smart Server .................................................................. 856
71.14 Unsuspend Smart Server .............................................................. 857
71.15 Unlock Smart Server .................................................................... 857
71.16 Shut down Smart Server ............................................................... 857
71.17 Stop Smart Server ....................................................................... 858
71.18 Open Smart Server Console ......................................................... 858
71.19 Smart Server Autoscaling .............................................................. 858
71.19.1 Get the List of Autoscaling Rules for Smart Server ...................... 859
71.19.2 Create Autoscaling Rule for SS .................................................. 860
71.19.3 Edit Autoscaling Rule for SS ....................................................... 860
71.19.4 Delete Autoscaling Rules ........................................................... 860
71.20 Smart Server Billing Statistics ...................................................... 861
71.21 Search Smart Servers by Label ..................................................... 861
71.22 Get Smart Server CPU Usage Statistics ....................................... 865
71.23 Resize Smart Server ..................................................................... 866
71.24 Add/Edit Admin/User Note for Smart Servers ............................... 867
71.25 Get List of Smart Server Blacklisted Domains ............................... 867
71.26 Edit Smart Server Blacklisted Domains ........................................ 868
71.27 Remove All Smart Server Domains from Blacklist ....................... 869

72 Software Licenses ........................................................................... 870
72.1 Get List of Software Licenses ......................................................... 870
72.2 Get Software License Details ......................................................... 871
72.3 Add Software License .................................................................... 872
72.4 Edit Software License .................................................................... 872
72.5 Delete Software License ................................................................. 873

73 SSH keys ......................................................................................... 874
73.1 Get List of SSH Keys ...................................................................... 874
73.2 Add SSH Key ................................................................................ 874
73.3 Edit SSH Key ................................................................................ 875
73.4 Delete SSH Key ............................................................................. 876

74 Statistics ......................................................................................... 877
74.1 Usage Trends Statistics ................................................................. 877
74.2 Cloud Usage Statistics ................................................................. 878

75 Storage Server Backups ................................................................. 880
75.1 Get All Storage Server Backups Details ......................................... 880
75.2 Get Normal Storage Server Backups Details .................................. 882
75.3 Get Incremental Server Backups Details ........................................ 884
75.4 Add Backup for Storage Server ..................................................... 886
75.5 Add/Edit Storage Server Backup Note ............................................ 888
75.6 Restore Storage Server Backup ..................................................... 888
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>76</td>
<td>System Configuration</td>
<td>890</td>
</tr>
<tr>
<td>76.1</td>
<td>View System Configuration</td>
<td>890</td>
</tr>
<tr>
<td>76.2</td>
<td>Edit System Configuration</td>
<td>902</td>
</tr>
<tr>
<td>77</td>
<td>Service Add-ons</td>
<td>915</td>
</tr>
<tr>
<td>77.1</td>
<td>Get List of Service Add-ons</td>
<td>915</td>
</tr>
<tr>
<td>77.2</td>
<td>Get Service Add-on Details</td>
<td>916</td>
</tr>
<tr>
<td>77.3</td>
<td>Create Service Add-on</td>
<td>917</td>
</tr>
<tr>
<td>77.4</td>
<td>Edit Service Add-on</td>
<td>918</td>
</tr>
<tr>
<td>77.5</td>
<td>Delete Service Add-on</td>
<td>919</td>
</tr>
<tr>
<td>78</td>
<td>Service Add-on Events</td>
<td>921</td>
</tr>
<tr>
<td>78.1</td>
<td>Get List of Service Add-on Events</td>
<td>921</td>
</tr>
<tr>
<td>78.2</td>
<td>Create Service Add-on Run Recipe Action</td>
<td>922</td>
</tr>
<tr>
<td>78.3</td>
<td>Create Service Add-on Raise Event Action</td>
<td>922</td>
</tr>
<tr>
<td>78.4</td>
<td>Edit Service Add-on Event</td>
<td>923</td>
</tr>
<tr>
<td>78.5</td>
<td>Delete Service Add-on Event</td>
<td>923</td>
</tr>
<tr>
<td>79</td>
<td>Service Add-on Groups</td>
<td>925</td>
</tr>
<tr>
<td>79.1</td>
<td>Get List of Service Add-on Groups</td>
<td>925</td>
</tr>
<tr>
<td>79.2</td>
<td>Get Service Add-on Group Details</td>
<td>927</td>
</tr>
<tr>
<td>79.3</td>
<td>Get Service Add-ons Attached to Service Add-on Group</td>
<td>928</td>
</tr>
<tr>
<td>79.4</td>
<td>Create Service Add-on Group</td>
<td>929</td>
</tr>
<tr>
<td>79.5</td>
<td>Add Service Add-on to Service Add-on Group</td>
<td>930</td>
</tr>
<tr>
<td>79.6</td>
<td>Edit Service Add-on Group</td>
<td>931</td>
</tr>
<tr>
<td>79.7</td>
<td>Edit Attached Service Add-on</td>
<td>931</td>
</tr>
<tr>
<td>79.8</td>
<td>Delete Service Add-on Group</td>
<td>932</td>
</tr>
<tr>
<td>79.9</td>
<td>Detach Service Add-on from Service Add-on Group</td>
<td>932</td>
</tr>
<tr>
<td>80</td>
<td>Templates</td>
<td>933</td>
</tr>
<tr>
<td>80.1</td>
<td>Get List of All Templates</td>
<td>933</td>
</tr>
<tr>
<td>80.2</td>
<td>Get List of System Templates</td>
<td>936</td>
</tr>
<tr>
<td>80.3</td>
<td>Get List of Own Templates</td>
<td>938</td>
</tr>
<tr>
<td>80.4</td>
<td>Get List of User Templates</td>
<td>941</td>
</tr>
<tr>
<td>80.5</td>
<td>Get List of Templates of Particular User</td>
<td>943</td>
</tr>
<tr>
<td>80.6</td>
<td>Get List of Inactive Templates</td>
<td>946</td>
</tr>
<tr>
<td>80.7</td>
<td>Get Template Details</td>
<td>948</td>
</tr>
<tr>
<td>80.8</td>
<td>Make Template Public</td>
<td>951</td>
</tr>
<tr>
<td>80.9</td>
<td>Delete Template</td>
<td>951</td>
</tr>
<tr>
<td>80.10</td>
<td>Edit Template</td>
<td>952</td>
</tr>
<tr>
<td>80.11</td>
<td>Get List of Available for Installation Templates</td>
<td>953</td>
</tr>
<tr>
<td>80.12</td>
<td>Get List of Available for Update Templates</td>
<td>955</td>
</tr>
<tr>
<td>80.13</td>
<td>Install Template</td>
<td>957</td>
</tr>
<tr>
<td>80.14</td>
<td>Update Template</td>
<td>961</td>
</tr>
<tr>
<td>80.15</td>
<td>Search for Particular Available for Installation Template</td>
<td>961</td>
</tr>
<tr>
<td>80.16</td>
<td>Search for Particular Available for Upgrade Template</td>
<td>963</td>
</tr>
<tr>
<td>81</td>
<td>Template Store</td>
<td>965</td>
</tr>
<tr>
<td>81.1</td>
<td>Get Template Store Details</td>
<td>965</td>
</tr>
<tr>
<td>81.2</td>
<td>Get Template Group Details</td>
<td>966</td>
</tr>
<tr>
<td>81.3</td>
<td>Add Template Group</td>
<td>967</td>
</tr>
<tr>
<td>81.4</td>
<td>Add Child Group</td>
<td>968</td>
</tr>
<tr>
<td>81.5</td>
<td>Edit Template Group</td>
<td>970</td>
</tr>
<tr>
<td>81.6</td>
<td>Delete Template Group</td>
<td>971</td>
</tr>
<tr>
<td>81.7</td>
<td>Get List of Templates Attached to Group</td>
<td>971</td>
</tr>
<tr>
<td>81.8</td>
<td>Attach Template to Group</td>
<td>973</td>
</tr>
<tr>
<td>81.9</td>
<td>Detach Template from Group</td>
<td>974</td>
</tr>
<tr>
<td>82</td>
<td>Top IOPS Disks</td>
<td>975</td>
</tr>
<tr>
<td>82.1</td>
<td>Get TOP IOPS Disks</td>
<td>975</td>
</tr>
<tr>
<td>83</td>
<td>Transaction Approvals</td>
<td>976</td>
</tr>
<tr>
<td>83.1</td>
<td>Get Approvals for Role</td>
<td>976</td>
</tr>
<tr>
<td>83.2</td>
<td>Get List of Transactions Pending Approval</td>
<td>976</td>
</tr>
</tbody>
</table>
83.3 Set Approvals for Role ................................................................. 977
83.4 Approve Transaction ................................................................... 978
83.5 Decline Transaction ................................................................... 978

84 Transactions .................................................................................. 980
84.1 Get List of Transactions .............................................................. 980
84.2 Get List of VS Transactions without Log Output ...................... 982
84.3 Get List of Transactions with Log Output ................................ 984
84.4 Get Transaction Details .............................................................. 986
84.5 Get Transaction Details Without Log Output ......................... 987

85 User Additional Fields .................................................................. 989
85.1 Get List of Additional Fields ..................................................... 989
85.2 Get Additional Field Details ...................................................... 989
85.3 Add Additional Field ................................................................. 990
85.4 Edit Additional Field ................................................................. 991
85.5 Delete Additional Field ............................................................. 991
85.6 Search User by Additional Field Parameter .......................... 992

86 User Groups .................................................................................... 993
86.1 Get List of User Groups ............................................................. 993
86.2 Get User Group Details .............................................................. 995
86.3 Get List of Users Assigned to User Group .............................. 997
86.4 Add User Group ....................................................................... 1001
86.5 Edit User Group ...................................................................... 1002
86.6 Delete User Group ................................................................... 1002

87 Users ............................................................................................. 1003
87.1 Get List of Users ....................................................................... 1003
87.2 Get List of Users (Short) .......................................................... 1007
87.3 Get User Details ..................................................................... 1010
87.4 Check Login/E-mail Availability ............................................. 1014
87.5 Add User ................................................................................ 1016
87.6 Edit User ............................................................................... 1017
87.7 Suspend User ......................................................................... 1018
87.8 Unlock User ........................................................................... 1019
87.9 Unlock User ........................................................................... 1019
87.10 Delete User ........................................................................... 1020
87.11 Drop Sessions .................................................................... 1020
87.12 Get User Last Access Log ....................................................... 1021
87.13 Get User Statistics ................................................................. 1022
87.14 Get User’s Statistics for Particular Period ............................... 1027
87.15 Get User Billing Statistics ...................................................... 1030
87.16 Get List of User Monthly Bills ............................................. 1032
87.17 Get List of User Payments ..................................................... 1032
87.18 Add Payment ...................................................................... 1033
87.19 Edit Payment ...................................................................... 1034
87.20 Delete Payment ..................................................................... 1034
87.21 Get List of User VSS .............................................................. 1035
87.22 Get List of Compute Resources Used by Users’ VSS ......... 1035
87.23 Get List of User Backups ......................................................... 1036
87.24 Get List of User Data Store Zones .................................... 1036
87.25 Get List of User Limits .......................................................... 1036
87.26 Get List of User Network Zones ........................................... 1039
87.27 Search User Backups .............................................................. 1039
87.28 Generate API Key ................................................................. 1042
87.29 Search Users by Name ........................................................... 1043
87.30 Get List of User’s YubiKeys .................................................. 1043
87.31 Add Yubikey to User .............................................................. 1044
87.32 Delete User Yubikey .............................................................. 1045

88 Users with Config Problems ....................................................... 1046
88.1 Get the list of users without roles ........................................... 1046
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>88.2</td>
<td>Get the list of users without timezones</td>
<td>1046</td>
</tr>
<tr>
<td>88.3</td>
<td>Get the list of users without user groups</td>
<td>1047</td>
</tr>
<tr>
<td>89</td>
<td>Version</td>
<td>1048</td>
</tr>
<tr>
<td>90</td>
<td>Virtual Servers</td>
<td>1049</td>
</tr>
<tr>
<td>90.1</td>
<td>Get List of VSs</td>
<td>1049</td>
</tr>
<tr>
<td>90.2</td>
<td>Get VS Details</td>
<td>1055</td>
</tr>
<tr>
<td>90.3</td>
<td>Get Statuses for all Virtual Servers</td>
<td>1060</td>
</tr>
<tr>
<td>90.4</td>
<td>Get Virtual Server Status</td>
<td>1061</td>
</tr>
<tr>
<td>90.5</td>
<td>Get VS Acceleration Status</td>
<td>1062</td>
</tr>
<tr>
<td>90.6</td>
<td>Add VS</td>
<td>1063</td>
</tr>
<tr>
<td>90.7</td>
<td>Add VS from OVA Template</td>
<td>1072</td>
</tr>
<tr>
<td>90.8</td>
<td>Add Instance Package VS</td>
<td>1080</td>
</tr>
<tr>
<td>90.9</td>
<td>Add VMware VS</td>
<td>1082</td>
</tr>
<tr>
<td>90.10</td>
<td>View Encrypted VS Password</td>
<td>1085</td>
</tr>
<tr>
<td>90.11</td>
<td>Build or Rebuild VS</td>
<td>1085</td>
</tr>
<tr>
<td>90.12</td>
<td>Edit VS</td>
<td>1086</td>
</tr>
<tr>
<td>90.13</td>
<td>Clone Virtual Server</td>
<td>1088</td>
</tr>
<tr>
<td>90.14</td>
<td>Change VS Owner</td>
<td>1089</td>
</tr>
<tr>
<td>90.15</td>
<td>Reset VS Root Password</td>
<td>1089</td>
</tr>
<tr>
<td>90.16</td>
<td>Set SSH Keys</td>
<td>1091</td>
</tr>
<tr>
<td>90.16.1</td>
<td>New API Request</td>
<td>1091</td>
</tr>
<tr>
<td>90.17</td>
<td>CPU Quota</td>
<td>1092</td>
</tr>
<tr>
<td>90.17.1</td>
<td>Get CPU Quota for Virtual Server</td>
<td>1092</td>
</tr>
<tr>
<td>90.17.2</td>
<td>Edit CPU Quota for Virtual Server</td>
<td>1092</td>
</tr>
<tr>
<td>90.18</td>
<td>Edit FQDN</td>
<td>1093</td>
</tr>
<tr>
<td>90.19</td>
<td>Migrate VS</td>
<td>1094</td>
</tr>
<tr>
<td>90.20</td>
<td>Full Migrate VS</td>
<td>1095</td>
</tr>
<tr>
<td>90.21</td>
<td>Migrate Multiple Virtual Servers</td>
<td>1096</td>
</tr>
<tr>
<td>90.22</td>
<td>Hot Migrate Disks</td>
<td>1097</td>
</tr>
<tr>
<td>90.23</td>
<td>Migrate VS from Xen to KVM</td>
<td>1097</td>
</tr>
<tr>
<td>90.24</td>
<td>Set VIP Status for VS</td>
<td>1098</td>
</tr>
<tr>
<td>90.25</td>
<td>Delete VS</td>
<td>1099</td>
</tr>
<tr>
<td>90.26</td>
<td>Start up VS</td>
<td>1100</td>
</tr>
<tr>
<td>90.27</td>
<td>Segregate VS</td>
<td>1100</td>
</tr>
<tr>
<td>90.28</td>
<td>Desegregate VS</td>
<td>1101</td>
</tr>
<tr>
<td>90.29</td>
<td>Reboot VS</td>
<td>1102</td>
</tr>
<tr>
<td>90.30</td>
<td>Get List of Blacklisted Domains</td>
<td>1102</td>
</tr>
<tr>
<td>90.31</td>
<td>Edit Blacklisted Domains</td>
<td>1103</td>
</tr>
<tr>
<td>90.32</td>
<td>Remove All Domains from Blacklist</td>
<td>1103</td>
</tr>
<tr>
<td>90.33</td>
<td>Purge File(s)</td>
<td>1104</td>
</tr>
<tr>
<td>90.34</td>
<td>Purge All Content</td>
<td>1105</td>
</tr>
<tr>
<td>90.35</td>
<td>Reboot VS in Recovery</td>
<td>1105</td>
</tr>
<tr>
<td>90.36</td>
<td>Reboot VS from ISO</td>
<td>1105</td>
</tr>
<tr>
<td>90.37</td>
<td>Boot VS from ISO</td>
<td>1106</td>
</tr>
<tr>
<td>90.38</td>
<td>Suspend VS</td>
<td>1106</td>
</tr>
<tr>
<td>90.39</td>
<td>Unlock VS</td>
<td>1107</td>
</tr>
<tr>
<td>90.40</td>
<td>Unsuspend VS</td>
<td>1107</td>
</tr>
<tr>
<td>90.41</td>
<td>Shut down VS</td>
<td>1107</td>
</tr>
<tr>
<td>90.42</td>
<td>Stop VS</td>
<td>1108</td>
</tr>
<tr>
<td>90.43</td>
<td>Open VS Console</td>
<td>1108</td>
</tr>
<tr>
<td>90.44</td>
<td>VS Autoscaling</td>
<td>1108</td>
</tr>
<tr>
<td>90.44.1</td>
<td>Enable Autoscaling for VS</td>
<td>1109</td>
</tr>
<tr>
<td>90.44.2</td>
<td>Get the List of Autoscaling Rules for VS</td>
<td>1109</td>
</tr>
<tr>
<td>90.44.3</td>
<td>Create Autoscaling Rule for VS</td>
<td>1110</td>
</tr>
<tr>
<td>90.44.4</td>
<td>Edit Autoscaling Rule for VS</td>
<td>1111</td>
</tr>
<tr>
<td>90.44.5</td>
<td>Delete Autoscaling Rule</td>
<td>1111</td>
</tr>
<tr>
<td>90.44.6</td>
<td>Disable Autoscaling for VS</td>
<td>1112</td>
</tr>
<tr>
<td>90.45</td>
<td>Get VS Billing Statistics</td>
<td>1112</td>
</tr>
<tr>
<td>90.46</td>
<td>Search VS by Label</td>
<td>1117</td>
</tr>
<tr>
<td>90.47</td>
<td>Get VS CPU Usage Statistics</td>
<td>1117</td>
</tr>
<tr>
<td>Page</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>90.48</td>
<td>Add/Edit Admin/User Note for Virtual Server</td>
<td></td>
</tr>
<tr>
<td>90.49</td>
<td>Enable Booting from CD for ISO Virtual Server</td>
<td></td>
</tr>
<tr>
<td>90.50</td>
<td>Disable Booting from CD for ISO Virtual Server</td>
<td></td>
</tr>
<tr>
<td>90.51</td>
<td>Get List of Service Add-ons Assigned to VS</td>
<td></td>
</tr>
<tr>
<td>90.52</td>
<td>Assign Service Add-on to VS</td>
<td></td>
</tr>
<tr>
<td>90.53</td>
<td>Unassign Service Add-on from VS</td>
<td></td>
</tr>
<tr>
<td>90.54</td>
<td>Use VS as Gateway</td>
<td></td>
</tr>
<tr>
<td>90.55</td>
<td>Virtual Server XML Config</td>
<td></td>
</tr>
<tr>
<td>90.55.1</td>
<td>Get VS XML Config</td>
<td></td>
</tr>
<tr>
<td>90.55.2</td>
<td>Edit VS XML Config</td>
<td></td>
</tr>
<tr>
<td>90.55.3</td>
<td>Reset VS XML Config</td>
<td></td>
</tr>
<tr>
<td>90.56</td>
<td>Virtual Server Backup Resources</td>
<td></td>
</tr>
<tr>
<td>90.56.1</td>
<td>Get List of Virtual Server Backup Resources</td>
<td></td>
</tr>
<tr>
<td>90.56.2</td>
<td>Add Backup Resource to Virtual Server</td>
<td></td>
</tr>
<tr>
<td>90.56.3</td>
<td>Remove Backup Resource from Virtual Server</td>
<td></td>
</tr>
<tr>
<td>90.57</td>
<td>Virtual Server Recovery Points</td>
<td></td>
</tr>
<tr>
<td>90.57.1</td>
<td>Create Recovery Point</td>
<td></td>
</tr>
<tr>
<td>90.57.2</td>
<td>Get List of Virtual Server Recovery Points</td>
<td></td>
</tr>
<tr>
<td>90.57.3</td>
<td>Get Recovery Point Details</td>
<td></td>
</tr>
<tr>
<td>90.57.4</td>
<td>Get List of File Entries for Recovery Point</td>
<td></td>
</tr>
<tr>
<td>90.57.5</td>
<td>Restore Virtual Server from Recovery Point</td>
<td></td>
</tr>
<tr>
<td>90.57.6</td>
<td>Restore File Entries from Recovery Point</td>
<td></td>
</tr>
<tr>
<td>90.58</td>
<td>Get Virtual Server Max Memory</td>
<td></td>
</tr>
<tr>
<td>90.59</td>
<td>Edit Virtual Server Max Memory</td>
<td></td>
</tr>
<tr>
<td>90.60</td>
<td>Enable Virsh Console</td>
<td></td>
</tr>
<tr>
<td>90.61</td>
<td>Disable Virsh Console</td>
<td></td>
</tr>
<tr>
<td>90.62</td>
<td>Add/Edit OVA VS License</td>
<td></td>
</tr>
<tr>
<td>90.63</td>
<td>Add/Edit OVA VS Config</td>
<td></td>
</tr>
<tr>
<td>91</td>
<td>Virtual Routers</td>
<td></td>
</tr>
<tr>
<td>91.1</td>
<td>Get List of Virtual Routers</td>
<td></td>
</tr>
<tr>
<td>91.2</td>
<td>Get Virtual Router Details</td>
<td></td>
</tr>
<tr>
<td>91.3</td>
<td>Convert Virtual Server to Virtual Router</td>
<td></td>
</tr>
<tr>
<td>91.4</td>
<td>Get List of IP Nets Assigned to Virtual Router</td>
<td></td>
</tr>
<tr>
<td>91.5</td>
<td>Get List of Attachable IP Nets</td>
<td></td>
</tr>
<tr>
<td>91.6</td>
<td>Assign IP Net to Virtual Router</td>
<td></td>
</tr>
<tr>
<td>91.7</td>
<td>Unassign IP Net from Virtual Router</td>
<td></td>
</tr>
<tr>
<td>92</td>
<td>Whitelist IPs</td>
<td></td>
</tr>
<tr>
<td>92.1</td>
<td>Get List of Whitelist IPs</td>
<td></td>
</tr>
<tr>
<td>92.2</td>
<td>Get Whitelist IP Details</td>
<td></td>
</tr>
<tr>
<td>92.3</td>
<td>Add Whitelisted IP</td>
<td></td>
</tr>
<tr>
<td>92.4</td>
<td>Edit Whitelisted IP</td>
<td></td>
</tr>
<tr>
<td>92.5</td>
<td>Delete Whitelisted IP</td>
<td></td>
</tr>
<tr>
<td>93</td>
<td>Zabbix Server</td>
<td></td>
</tr>
<tr>
<td>93.1</td>
<td>Zabbix Server Setup</td>
<td></td>
</tr>
<tr>
<td>93.2</td>
<td>Reconfigure Existing Zabbix Server</td>
<td></td>
</tr>
</tbody>
</table>

---

90.48 Add/Edit Admin/User Note for Virtual Server................................................................1118
90.49 Enable Booting from CD for ISO Virtual Server..............................................................1120
90.50 Disable Booting from CD for ISO Virtual Server..............................................................1120
90.51 Get List of Service Add-ons Assigned to VS.................................................................1121
90.52 Assign Service Add-on to VS.......................................................................................1122
90.53 Unassign Service Add-on from VS..................................................................................1123
90.54 Use VS as Gateway........................................................................................................1123
90.55 Virtual Server XML Config........................................................................................1124
90.55.1 Get VS XML Config......................................................................................................1124
90.55.2 Edit VS XML Config......................................................................................................1124
90.55.3 Reset VS XML Config......................................................................................................1125
90.56 Virtual Server Backup Resources..................................................................................1125
90.56.1 Get List of Virtual Server Backup Resources.............................................................1125
90.56.2 Add Backup Resource to Virtual Server........................................................................1127
90.56.3 Remove Backup Resource from Virtual Server...............................................................1128
90.57 Virtual Server Recovery Points......................................................................................1128
90.57.1 Create Recovery Point..................................................................................................1128
90.57.2 Get List of Virtual Server Recovery Points.................................................................1129
90.57.3 Get Recovery Point Details...........................................................................................1130
90.57.4 Get List of File Entries for Recovery Point.................................................................1131
90.57.5 Restore Virtual Server from Recovery Point..................................................................1132
90.57.6 Restore File Entries from Recovery Point......................................................................1133
90.58 Get Virtual Server Max Memory.......................................................................................1134
90.59 Edit Virtual Server Max Memory......................................................................................1135
90.60 Enable Virsh Console........................................................................................................1135
90.61 Disable Virsh Console.......................................................................................................1136
90.62 Add/Edit OVA VS License..................................................................................................1136
90.63 Add/Edit OVA VS Config.....................................................................................................1137

---

91 Virtual Routers....................................................................................................................1138
91.1 Get List of Virtual Routers...............................................................................................1138
91.2 Get Virtual Router Details...............................................................................................1143
91.3 Convert Virtual Server to Virtual Router..........................................................................1147
91.4 Get List of IP Nets Assigned to Virtual Router..................................................................1148
91.5 Get List of Attachable IP Nets...........................................................................................1149
91.6 Assign IP Net to Virtual Router..........................................................................................1150
91.7 Unassign IP Net from Virtual Router..................................................................................1150

92 Whitelist IPs........................................................................................................................1152
92.1 Get List of Whitelist IPs...................................................................................................1152
92.2 Get Whitelist IP Details.....................................................................................................1153
92.3 Add Whitelisted IP..............................................................................................................1153
92.4 Edit Whitelisted IP..............................................................................................................1154
92.5 Delete Whitelisted IP..........................................................................................................1154

93 Zabbix Server.......................................................................................................................1156
93.1 Zabbix Server Setup............................................................................................................1156
93.2 Reconfigure Existing Zabbix Server.....................................................................................1156
The API enables cloud integration with third party applications. This guide is a complete reference for all API calls and includes detailed API information, code and output examples. The version of the guide corresponds to the latest OnApp API version. For comprehensive instructions on previous versions, refer to corresponding guides at docs.onapp.com.

- The OnApp API is RESTful
- All function calls respond to XML and JSON exchange formats
- All function calls need authorization and authentication (Basic HTTP or API key)
- The OnApp API is backward compatible within one major version. However, a new major version might include changes that are not backward compatible with the previous one.
1 API Authentication

To authenticate using HTTP Basic, just use your username/password combination. Curl example:

```
curl -u user:userpass
```

To authenticate using API key, put your account email as a login and the key to the server as a password.
2 HTTP Methods

The API uses the following HTTP methods:

**GET** - used for retrieving information from a particular URI

**POST** - used for creating new object and adding new transactions into the queue

**PUT** - used for altering object properties

**NOTE:** the `updated_at` value is changed in PUT requests even if the request fails.

**DELETE** - used for object deletion
## 3 HTTP Response Codes

The API returns appropriate HTTP status codes for every request:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200 OK</td>
<td>The request completed successfully</td>
</tr>
<tr>
<td>204 No content</td>
<td>The request completed successfully. The 204 status is returned on DELETE and PUT requests</td>
</tr>
<tr>
<td>201 Scheduled</td>
<td>The request has been accepted and scheduled for processing</td>
</tr>
<tr>
<td>403 Forbidden</td>
<td>The request is correct, but could not be processed.</td>
</tr>
<tr>
<td>404 Not Found</td>
<td>The requested URL is incorrect or the resource does not exist. For example, if you request to delete a user with ID 5, but there is no such a user in the cloud, you will get a 404 error.</td>
</tr>
<tr>
<td>422 Unprocessable Entity</td>
<td>The sent parameters are erroneous.</td>
</tr>
<tr>
<td>500 Internal Server Error</td>
<td>An error occurred. Please contact support.</td>
</tr>
<tr>
<td>503 Service Unavailable</td>
<td>The request cannot be handled currently, due to a temporary overloading or maintenance of the server. This condition is temporary and the request will be handed after a certain delay.</td>
</tr>
</tbody>
</table>
4 Formatting and Naming Conventions

The table below represents all the existing formatting and naming conventions used in this guide:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Explanation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>user:userpass</td>
<td>stands for username:password combination</td>
<td>Admin:123456</td>
</tr>
<tr>
<td>onapp.test</td>
<td>stands for address, where your Control Panel is located</td>
<td>Example.com</td>
</tr>
<tr>
<td>:id</td>
<td>stands for the resource ID. Sometimes also: :resource_id</td>
<td>23</td>
</tr>
<tr>
<td>italics</td>
<td>all the parameters are italicised</td>
<td>currency_code</td>
</tr>
<tr>
<td>* (asterisk)</td>
<td>marks the required parameters</td>
<td>label *</td>
</tr>
<tr>
<td>preformatted</td>
<td>indicates request examples in XML or JSON</td>
<td>GET /roles.xml</td>
</tr>
<tr>
<td>Code block</td>
<td>indicates console requests and response examples.</td>
<td></td>
</tr>
</tbody>
</table>
```xml
<?xml version="1.0" encoding="UTF-8"?>
<service_addon><id>2</id><label>service_addon2</label></service_addon>
``` |
<p>| info | An info message emphasizes or explains the information within the chapter. | Clicking the OFF button performs graceful shutdown and then powers off the VS. |
| note | A note message contains information essential for the task | The maximum length of a Mount Point is 256 characters. |</p>
<table>
<thead>
<tr>
<th><strong>warning</strong></th>
<th>A warning message informs you of something you should not do or be cautious.</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://via.placeholder.com/15" alt="warning" /></td>
<td>You won't be able to restore a VS after deleting it.</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/15" alt="tip" /></td>
<td>The element showing new parameters added in the latest release of API.</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/15" alt="tip" /></td>
<td>limit_type – hourly or monthly limit type set for the resource</td>
</tr>
</tbody>
</table>
5 FAQ

Q: Is it possible to enable API access via HTTPS?
A: We can enable https for your cloud, which can be used for both WebUI access and API access. Or you can do so yourself: the Apache config file is located at: /etc/httpd/conf.d/onapp.conf

Q: Can I create a VS on behalf of another user?
A: No, you can't but you can switch the VS owners. Refer to Change a VS owner section for details.

Q: How are passwords stored – in plain text?
A: No, passwords are not stored in plain text. Except for a login and password combination, you can use email + API key combination to authorize a user via the API. API keys can be generated and changed easily on a user's profile page (as well as through the API). For security reasons we recommend users authenticate through the API key, not the login and password.

Q: Which parameters are required and which are optional?
A: Required parameters are marked in this guide with an asterisk *.
6 About API Guide

OnApp API enables you to integrate cloud with third party applications. This guide provides all available API requests and includes detailed API information, code, and examples.

- The OnApp API is RESTful
- All function calls respond to XML and JSON exchange formats
- All function calls need authorization and authentication (Basic HTTP or API key)
- The OnApp API is backward compatible within one major version. However, a new major version might include changes that are not backward compatible with the previous one.

6.1 API Authentication

To authenticate using HTTP Basic, just use your username/password combination. Curl example:

```
curl -u user:use
```

To authenticate using API key, put your account email as a login and the key to the server as a password.

6.2 HTTP Methods

The API uses the following HTTP methods:

- **GET** - used for retrieving information from a particular URI
- **POST** - used for creating new object and adding new transactions into the queue
- **PUT** - used for altering object properties

**NOTE:** the updated_at value is changed in PUT requests even if the request fails.

- **DELETE** - used for object deletion

6.3 HTTP Response Codes

The API returns appropriate HTTP status codes for every request:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>The request completed successfully</td>
</tr>
<tr>
<td>204</td>
<td>The request completed successfully. The 204 status is returned on DELETE and PUT requests</td>
</tr>
<tr>
<td>201</td>
<td>The request has been accepted and scheduled for processing</td>
</tr>
<tr>
<td>403</td>
<td>The request is correct, but could not be processed.</td>
</tr>
</tbody>
</table>
404 Not Found

The requested URL is incorrect or the resource does not exist. For example, if you request to delete a user with ID {5}, but there is no such a user in the cloud, you will get a 404 error.

422 Unprocessable Entity

The sent parameters are erroneous.

500 Internal Server Error

An error occurred. Please contact support.

503 Service Unavailable

The request cannot be handled currently, due to a temporary overloading or maintenance of the server. This condition is temporary and the request will be handed after a certain delay.

6.4 Formatting and Naming Conventions

The table below represents all the existing formatting and naming conventions used in this guide:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Explanation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>user:userpassword</td>
<td>stands for username:password combination</td>
<td>Admin:123456</td>
</tr>
<tr>
<td>onapp.test</td>
<td>stands for address, where your Control Panel is located</td>
<td>Example.com</td>
</tr>
<tr>
<td>:id</td>
<td>stands for the resource ID. Sometimes also: :resource_id</td>
<td>23</td>
</tr>
<tr>
<td>italics</td>
<td>all the parameters are italicised</td>
<td>currency_code</td>
</tr>
<tr>
<td>* (asterisk)</td>
<td>marks the required parameters</td>
<td>label *</td>
</tr>
<tr>
<td>preformatted</td>
<td>indicates request examples in XML or JSON</td>
<td>GET /roles.xml</td>
</tr>
<tr>
<td>Code</td>
<td>Code block indicates console requests and response examples.</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>info</td>
<td>An info message emphasizes or explains the information within the chapter.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Clicking the OFF button performs graceful shutdown and then powers off the VS.</td>
<td></td>
</tr>
<tr>
<td>note</td>
<td>A note message contains information essential for the task completion.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The maximum length of a Mount Point is 256 characters.</td>
<td></td>
</tr>
<tr>
<td>war ning</td>
<td>A warning message informs you of something you should not do or be cautious.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>You won't be able to restore a VS after deleting it.</td>
<td></td>
</tr>
<tr>
<td>🕯️</td>
<td>The element showing new parameters added in the latest release of API.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$limit_type$ – hourly or monthly limit type set for the resource</td>
<td></td>
</tr>
</tbody>
</table>

### 6.5 FAQ

**Q: Is it possible to enable API access via HTTPS?**

A: We can enable https for your cloud, which can be used for both WebUI access and API access. Or you can do so yourself: the Apache config file is located at: `/etc/httpd/conf.d/onapp.conf`

**Q: Can I create a VS on behalf of another user?**

A: No, you can't but you can switch the VS owners. Refer to [Change a VS owner](#) section for details.

**Q: How are passwords stored – in plain text?**

A: No, passwords are not stored in plain text. Except for a login and password combination, you can use email + API key combination to authorize a user via the API. API keys can be generated and changed easily on a user's profile page (as well as through the API). For security reasons we recommend users authenticate through the API key, not the login and password.

**Q: Which parameters are required and which are optional?**
A: Required parameters are marked in this guide with an asterisk *. 

7 Accelerator

Accelerator dashboard is aimed at enabling acceleration for all types of networks to speed up the traffic flow running for the corresponding virtual server. In this section, you can view the list of accelerated networks, allow and prohibit acceleration for the networks.

7.1 Get List of Accelerated Networks

To view accelerated networks, use the following request:

GET /settings/cdn_network_accelerations.xml
GET /settings/cdn_network_accelerations.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<networks type="array">
  <network>
    <id type="integer">1</id>
    <label>public</label>
    <network_acceleration_id nil="true"/>
    <accelerator_id nil="true"/>
  </network>
  <network>
    <id type="integer">2</id>
    <label>private</label>
    <network_acceleration_id nil="true"/>
    <accelerator_id nil="true"/>
  </network>
  ...
</networks>
```

Where:

- **id** - the ID of the network
- **label** - network name
- **network_acceleration_id** - the ID of the accelerated network
- **accelerator_id** - the ID of the accelerator. True if the accelerator is built, otherwise false.
7.2 Enable Network Acceleration

To enable network acceleration, use the following request:

POST /settings/cdn_network_accelerations.xml
POST /settings/cdn_network_accelerations.json

**XML Request Example**

```
curl "http://onapp.test/settings/cdn_network_accelerations.xml" -d
' <network_id>6</network_id>' -X POST \
  -u user:userpass \ 
  -H "Accept: application/xml" \ 
  -H "Content-Type: application/xml"
```

**JSON Request Example**

```
curl "http://onapp.test/settings/cdn_network_accelerations.json" -d
'{"network_id":6}' -X POST \
  -u user:userpass \ 
  -H "Accept: application/json" \ 
  -H "Content-Type: application/json"
```

Where:

*network_id* - the ID of the network to be accelerated

7.3 Disable Network Acceleration

To disable network acceleration, use the following request:

DELETE /settings/cdn_network_accelerations/:id.xml
DELETE /settings/cdn_network_accelerations/:id.json

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url
  http://onapp.test/settings/cdn_network_accelerations/4.xml
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass --url
  http://onapp.test/settings/cdn_network_accelerations/4.json
```

7.4 Enable Accelerator for Virtual Server

To allow acceleration for VS, use the following request:

PUT /virtual_machines/:id/acceleration/enable.xml
PUT /virtual_machines/:id/acceleration/enable.json

The following requests are still valid but will be deprecated in the future releases:
POST /virtual_machines/:id/accelerate.xml
POST /virtual_machines/:id/accelerate.json

XML Request Example

curl -i -X PUT
http://onapp.test/virtual_machines/hxgczpptestetr/acceleration/enable.xml
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X PUT
http://onapp.test/virtual_machines/hxgczpptestetr/acceleration/enable.json
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

7.5 Disable Accelerator for Virtual Server

To prohibit acceleration for VS, use the following request:

PUT /virtual_machines/:id/acceleration/disable.xml
PUT /virtual_machines/:id/acceleration/disable.json

The following requests are still valid but will be deprecated in the future releases:

POST /virtual_machines/:id/decelerate.xml
POST /virtual_machines/:id/decelerate.json

XML Request Example

curl -i -X PUT
http://onapp.test/virtual_machines/hxgczpptestetr/acceleration/disable.xml
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X PUT
http://onapp.test/virtual_machines/hxgczpptestetr/acceleration/disable.json
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

7.6 Enable Acceleration for Smart Server

To allow acceleration for a smart server, use the following request:

PUT /smart_servers/:id/acceleration.enable.xml
PUT /smart_servers/:id/acceleration.enable.json

The following requests are still valid but will be deprecated in the future releases:

POST /smart_servers/:id/accelerate.xml
POST /smart_servers/:id/accelerate.json
XML Request Example

```
curl -i -X PUT
http://onapp.test/smart_servers/hxgczpstestet/acceleration.enable.xml -u
user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X PUT
http://onapp.test/smart_servers/hxgczpstestet/acceleration.enable.json -u
user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

### 7.7 Disable Acceleration for Smart Server

To prohibit acceleration for a smart server, use the following request:

PUT `/smart_servers/:id/acceleration.disable.xml`  
PUT `/smart_servers/:id/acceleration.disable.json`

The following requests are still valid but will be deprecated in the future releases:

POST `/smart_servers/:id/decelerate.xml`  
POST `/smart_servers/:id/decelerate.json`

XML Request Example

```
curl -i -X PUT
http://onapp.test/smart_servers/hxgczpstestet/acceleration.disable.xml -u
user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X PUT
http://onapp.test/smart_servers/hxgczpstestet/acceleration.disable.json -u
user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```
8 Accelerator Statistics

If you want to track the amount of bandwidth used by accelerated websites, you can view bandwidth statistics using API. By default, statistics are generated for the last 24 hours. This section contains the API requests you can use to get Accelerator statistics.

8.1 Get Summary Charts

Summary charts show detailed information on the total number of active websites that are currently accelerated by the accelerator.

To view statistics for the selected period, use the following request:

GET /acceleration_dashboard/summary_charts.xml
GET /acceleration_dashboard/summary_charts.json

XML Request Example

```
curl -G -X GET -u user:userpass --url
--data-urlencode 'start_date=2020-02-17 09:00', --data-urlencode 'end_date=2020-02-25 09:00' --data-urlencode 'frequency=1'
```

XML Output Example

```
<summary_charts type="array">
  <summary_chart>
    <timestamp type="integer">1567296000000</timestamp>
    <website_count type="integer">42</website_count>
  </summary_chart>
  <summary_chart>
    <timestamp type="integer">1582329600000</timestamp>
    <website_count type="integer">6</website_count>
  </summary_chart>
  ...</summary_charts>
```

Where:

- **timestamp** - time when the website count was collected in UTC format
- **website_count** - the total number of active websites that are currently accelerated by the accelerator
8.2 Get Summary Metrics

Summary metrics show detailed information on the amount of bandwidth used by accelerated websites.

To view statistics for the selected period, use the following request:

GET /acceleration_dashboard/summary_metrics.xml
GET /acceleration_dashboard/summary_metrics.json

XML Request Example

```
curl -G -X GET -u user:usertest --url http://onapp.test/acceleration_dashboard/summary_metrics.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' --data-urlencode 'start_date=2020-02-16 09:00', --data-urlencode 'end_date=2020-02-17 09:00'
```

JSON Request Example

```
```

XML Output Example

```
<summary_metrics>
  <highest_cdn_bandwidth type="float">1905336.0</highest_cdn_bandwidth>
  <highest_origin_bandwidth type="float">895136494.0</highest_origin_bandwidth>
</summary_metrics>
```

Where:

- **highest_cdn_bandwidth** - the highest total cached bandwidth (traffic served from CDN) served per day within the selected period
- **highest_origin_bandwidth** - the highest total uncached bandwidth (traffic served from the origin) served per day within the selected period
9 Add Zabbix Server

To add a Zabbix server to your system, use the following request:

POST /sysadmin_tools/infrastructure/zabbix_setup.xml
POST /sysadmin_tools/infrastructure/zabbix_setup.json

If you already have a Zabbix server, you can connect it to your cloud. For more information, see Edit System Configuration.

XML Request Example

curl -i -X POST
http://onapp.test/sysadmin_tools/infrastructure/zabbix_setup.xml -d
'<?xml version="1.0" encoding="UTF-8"?>
<zabbix_setup>
  <ip_address>1.2.3.4</ip_address>
</zabbix_setup>'
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X POST
http://onapp.test/sysadmin_tools/infrastructure/zabbix_setup.json -d
'{"zabbix_setup":{"ip_address": "1.2.3.4"}}'
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

Where:

*ip_address* - the IP address of the server on which OnApp will automatically configure Zabbix. It can be either a physical server or a virtual one.
10 Alerts

To get the list of alerts on Zombie disks, Zombie data stores, Zombie virtual servers, and Zombie transactions, use the following request:

GET /alerts.xml
GET /alerts.json

XML Request Example


JSON Request Example


XML Output Example

<alerts>
<zombie_data_stores type="array"/>
<zombie_transactions type="array"/>
<zombie_disks type="array">
<zombie_disk>/dev/[DATASTORE_IDENTIFIER]/[DISK_IDENTIFIER]</zombie_disk>
</zombie_disks>
<zombie_domains type="array">
<zombie_domain>[VM_IDENTIFIER]</zombie_domain>
</zombie_domains>
</alerts>

Where:

- **zombie_data_stores** - data stores, which are not mounted to a compute resource
- **zombie_transactions** - transactions, which have "running" status, but their PIDs do not exist in the system
- **zombie_disks** - disks, which exist on data stores, but are not in the database
- **zombie_domains** - an array of zombie VSSs, which are running on a compute resource, but are not in the DB
11 Application Servers

Application Server is a regular VS based on default CentOS template with pre-installed additional software. This software allows you to install and have up & running various PHP/Perl/Python frameworks (like Drupal, Joomla, Wordpress etc.) on a server using web interface.

API requests for application servers are the same as for virtual servers, just make the following replacement:

```
virtual_machines->application_servers
```

Below you can find the list of operations applicable for application servers:

- Get List of servers
- Get server details
- Get statuses for all servers
- Get server status
- Add server
- Build server
- Edit server
- Change server owner
- Migrate server
- Set VIP status for server
- Delete server
- Start up server
- Segregate server
- Reboot server
- Reboot server in recovery
- Suspend server
- Unlock server
- Unsuspend server
- Shut down server
- Stop server
- Server autoscaling
- Server billing statistics
- Search server by label
- Get server CPU usage statistics
- Add/edit admin/user note for server
11.1 Create Application Server

To create an application server, use the following request:

POST /application_servers.xml
POST /application_servers.json

XML Request Example

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<application_server><location_id>1</location_id><label>zaza</label><hostname>zaza</hostname><domain>localdomain</domain><hypervisor_group_id>6</hypervisor_group_id><hypervisor_id>2</hypervisor_id><memory>384</memory><cpus>1</cpus><cpu_shares>1</cpu_shares><data_store_group_primary_id>1</data_store_group_primary_id><primary_disk_size>5</primary_disk_size><data_store_group_swap_id>1</data_store_group_swap_id><swap_disk_size>1</swap_disk_size><primary_network_group_id>16</primary_network_group_id><required_ip_address_assignment>1</required_ip_address_assignment><rate_limit>0</rate_limit><user_group_id>2500</user_group_id><vdc_id>192</vdc_id><data_store_id>236</data_store_id><network_id>653</network_id><required_virtual_machine_build>1</required_virtual_machine_build></application_server>' --url http://onapp.test/application_servers.xml
```

JSON Request Example

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"application_server":{"location_id":"1", "label":"zaza JSON", "hostname":"zaza", "domain":"localdomain", "hypervisor_group_id":"6", "hypervisor_id":"2", "memory":"384", "cpus":"1", "cpu_shares":"1", "data_store_group_primary_id":"1", "primary_disk_size":"5", "data_store_group_swap_id":"1", "swap_disk_size":"1", "primary_network_group_id":"16", "user_group_id":"2500", "vdc_id":"192", "data_store_id":"236", "network_id":"653", "required_ip_address_assignment":"1", "rate_limit":"0", "required_virtual_machine_build":"1"}}' --url http://onapp.test/application_servers.json
```

XML Output Example

```xml
<application_server>
  <location_id>1</location_id>
  <label>zaza</label>
  <hostname>zaza</hostname>
  <domain>localdomain</domain>
  <hypervisor_group_id>6</hypervisor_group_id>
  <hypervisor_id>2</hypervisor_id>
  <memory>384</memory>
  <cpus>1</cpus>
  <cpu_shares>1</cpu_shares>
  <data_store_group_primary_id>1</data_store_group_primary_id>
  <primary_disk_size>5</primary_disk_size>
  <data_store_group_swap_id>1</data_store_group_swap_id>
  <swap_disk_size>1</swap_disk_size>
  <primary_network_group_id>16</primary_network_group_id>
  <required_ip_address_assignment>1</required_ip_address_assignment>
  <rate_limit>0</rate_limit>
  <user_group_id>2500</user_group_id>
  <vdc_id>192</vdc_id>
  <data_store_id>236</data_store_id>
  <network_id>653</network_id>
  <required_virtual_machine_build>1</required_virtual_machine_build>
</application_server>
```
<id type="integer">340</id>
<ip_address_pool_id nil="true"/>
<network_address>69.168.237.0</network_address>
<network_id type="integer">3</network_id>
<pxe type="boolean">false</pxe>
<updated_at type="datetime">2015-06-23T15:33:37+03:00</updated_at>
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>255.255.255.0</netmask>
</ip_addresses>
<monthly_bandwidth_used>0</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>
<price_per_hour type="float">769.0</price_per_hour>
<price_per_hour_powered_off type="float">384.01</price_per_hour_powered_off>
<support_incremental_backups type="boolean">true</support_incremental_backups>
<cpu_priority type="integer">1</cpu_priority>
</application_server>

Where:

* add_to_marketplace – true, if the server is added to marketplace. The default value is "false". This parameter is for CDN servers only.

* admin_note - administrator comment for the VS

* allowed_hot_migrate – true if hot migration is allowed, otherwise false

* allowed_swap – true if swap is allowed, otherwise false

* booted - true, if the VS is booted, otherwise false

* built - true, if the VS is built, otherwise false

* cpuShares - the percentage of allocated CPU priority resource

* cpu_sockets - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted

* cpu_threads - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted

* cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in a bucket.

* cpus - number of CPUs assigned to the VS

* created_at – the date when the VS was created in the [YYYY][MM][DD][hh][mm][ss] format

* updated_at – the date when the VS was updated in the [YYYY][MM][DD][hh][mm][ss] format

* customer_network_id - ID of the customer network

* domain - specify the domain for this VS. The default value is *localdomain*. This parameter is not applicable for Windows virtual servers.

* edge_server_type - type of the CDN edge server. This parameter is for CDN servers only.

* enable_autoscale — true if autoscaling is allowed for this VS

* enable_monitis - deprecated attribute

* firewall notrack - parameter for adding firewall rules. It is true for edge servers only.

* hostname - VS hostname

* hypervisor_id – the ID of the compute resource, on which the server is deployed
hypervisor_group_id – the ID of the compute zone, on which the server is deployed
id – the VS ID in OnApp CP database
identifier – the VS identifier
instance_package_id - ID of the instance package
iso_id - ID of the ISO you want to use
label - user-friendly VS description
local_remote_access_ip_address - IP address for remote connection
local_remote_access_port - port for remote connection
locked - true if the VS is locked; otherwise false
memory - the RAM size allocated to this VS
min_disk_size — the minimum disk size required to build a VS from a specified template
note - optional note
operating_system — operating system used by the VS
operating_system_distro — the distribution of the OS from which this VS is built
preferred_hvs - the array of preferable compute resources based on compute zone that meet some VS configuration settings
rate_limit - the port speed. The minimum recommended speed is 10Mbps (if allowed by your bucket)
recovery_mode - true if recovery mode allowed, otherwise false
remote_access_password — the password for the remote access
service_password - password of a service user
state – application server state
storage_server_type - set http or streaming server type.
strict_virtual_machine_id - the ID of a virtual machine that will never reside on the same compute resource with this VS
suspended - true if VS is suspended, otherwise false
template_id - the ID of the template the VS is based on
template_label - the name of the template from which this VS is built
time_zone – the time zone of the user
user_id — the ID of a user assigned to this VS
vip — true if the VS has VIP status (gives migration priority)
xen_id — the VS ID set by the virtualization engine
ip_addresses - an array of IP addresses with the following parameters:
  - address - IP address
  - broadcast - broadcast address
  - created_at - the date when the IP address was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - customer_network_id - ID of the customer network which will be used for this blueprint
  - disallowed_primary - true if not allowed to be used as primary (for VS build), otherwise false
  - gateway - gateway address
  - hypervisor id - the ID of the compute resource
- id - the ID of the IP address
- ip_address_pool_id - ID of the IP address pool to which the IP address belongs to
- network_address - the address of the network
- network_id - the ID of the network
- pxe - true, if this compute resource address can be used for cloudbooting a compute resource
- updated_at - the date when the IP address was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- user_id - the ID of a user associated with this IP address
- free - true if free, otherwise false
- netmask - netmask for the IP address

monthly_bandwidth_used - VS monthly bandwidth in KB

total_disk_size - total VS disk size

price_per_hour - server's price per hour

price_per_hour_powered_off - price per hour when server is powered off

support_incremental_backups - 1, if virtual server supports incremental backups, and 0 if it does not

cpu_priority - this is a new parameter reserved for further use; currently will have the same value as cpu_shares

Page History

v.5.4
- added domain parameter

v.4.2
- removed initial_root_password and initial_root_password_encrypted parameters as they are not required and skipped when creating an application server
12 Apps for Application Servers

An application is a piece of software that brings additional features into the basic functionality. OnApp allows you to deploy a wide range of applications by means of additional software. To install different applications on your cloud, you should create an Application Server. An application server is a regular VS based on CentOS but with pre-installed software. Application Servers allow you to install various applications (like Drupal, Joomla, Wordpress etc.) on a server using web interface.

12.1 Get List of All Installed Applications

To get the list of all installed applications, use the following request:

GET /application_servers/:application_server_id/applications.xml
GET /application_servers/:application_server_id/applications.json

**XML Request Example**

```plaintext
```

**JSON Request Example**

```plaintext
```

**XML Output Example**

```xml
<applications type="array">
  <application>
    <application_type>WordPress</application_type>
    <errors type="array"/>
    <id>26_74947</id>
    <software_url>http://109.1.125.2/wp</software_url>
    <software_version>4.2.2</software_version>
  </application>
</applications>
```

Where:

- `admin_url` - this URL is a link for administrator, where they can enter credentials to log into application
- `application_type` - the name of application
- `id` - the ID of an application
- `software_url` - this URL is a link to the application itself
- `software_version` - the version of the application software installed
12.2 Get List of All Available for Installation Applications

To get the list of all applications, available for installation, use the following request:

GET /application_servers/:application_server_id/applications/available.xml
GET /application_servers/:application_server_id/applications/available.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```xml
<applications type="array">
  <application>
    <category>cms</category>
    <description>Zikula is a Web Application Toolkit, which allows you to run impressive websites and build powerful online applications.</description>
    <name>Zikula</name>
    <script_id>1</script_id>
    <software_version>1.3.9</software_version>
  </application>
  <application>
    <category>forums</category>
    <description>The most widely used Open Source forum solution</description>
    <name>phpBB</name>
    <script_id>2</script_id>
    <software_version>3.1.4</software_version>
  </application>
  ...
</applications>
```

**Where:**

- **category** - the category, to which an application refers (for example - cms, forums, blogs, frameworks etc.)
- **description** - the short description of main application features
- **name** - the standard name of an application
- **script_id** - the ID of an application in the list of all applications, available for installation
- **software_version** - the version of application software
12.3 Get Application Attributes

To get attributes required for installation of an application and their default values, use the following request:

GET /application_servers/:application_server_id/script/:script_id.xml
GET /application_servers/:application_server_id/script/:script_id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

Where:

- `script_id` - the ID of an application in the list of all applications, available for installation. List of all available scripts can be extracted using request from the Get List of All Available for Installation Applications section. In this request you will get the following parameters:
  - `category` - the category, to which an application refers (for example - cms, forums, blogs, frameworks, etc.)
  - `description` - the short description of main application features
  - `name` - the standard name of an application
  - `script_id` - the ID of an application in the list of all applications, available for installation
  - `software_version` - the version of application software

12.4 Install Application

To install an application, use the following request:

POST /application_servers/:application_server_id/applications.xml
POST /application_servers/:application_server_id/applications.json

XML Request Example

```bash
curl -i -X POST -u user:userpass http://onapp.test/application_servers/109/applications.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<application><script_id>1</script_id><software_directory>XML_Zikula</software_directory><admin_username>admin</admin_username><admin_password>pass</admin_password><admin_email>user@onapp.com</admin_email></application>'
```

JSON Request Example
Where:

**script_id** - the ID of an application in the list of all applications, available for installation.

**softdirectory** - the directory where an application will be stored. The label of the directory should contain only lowercase characters.

The following parameters depend on the type of an application:

- **admin_username** - the username of an administrator, who wants to install an application
- **admin_pass** - the password of an administrator, who wants to install an application
- **admin_email** - the email of an administrator, who wants to install an application

Parameters are different for every application. All the parameters that are required for successful installation of the application can be retrieved via the [Get Application Attributes](#) API call. Put extracted parameters inside `<application> </application>` parameter.

### 12.5 Back Up Application

To back up an application, use the following request:

```
curl -i -X POST -u user:password  
http://onapp.test/application_servers/:application_server_id/applications/:id/backup.xml  
```

**XML Request Example**

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml'  
http://onapp.test/application_servers/:application_server_id/applications/:id/backup.xml  
-d  
'<?xml version='1.0' encoding='iso-8859-1'?>  
<backup>  
<application_id>1_31528</application_id>  
<backup_directory>1</backup_directory>  
<backup_database>1</backup_database>  
<note>xml backup</note>  
</backup>'
```

**JSON Request Example**

Where:

- application_id - the ID of an application
- backup_directory - set 1 to back up directory, otherwise set 0
- backup_data_directory - set 1 to back up data directory, otherwise set 0
- backup_database - set 1 to back up database, otherwise set 0
- note - a back up note (optional parameter)

Parameters backup_directory, backup_database can be used with any value (for example, "1" or "true") to back up a corresponding thing. Do not include a parameter in order not to back up particular part of an application.

12.6 Delete Application

To delete an application, use the following request:

DELETE /application_servers/:application_server_id/applications/:id.xml
DELETE /application_servers/:application_server_id/applications/:id.json

XML Request Example

curl -i -X DELETE -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password --url http://onapp.test/application_servers/109/applications/1_23728.xml -d '<application><application_id>1_31528</application_id><remove_directory>1</remove_directory><remove_database>1</remove_database><remove_database_user>1</remove_database_user></application>'

JSON Request Example

curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:password --url http://onapp.test/application_servers/109/applications/1_23728.json -d '{"application": {"application_id": "1_23728", "remove_directory": "1", "remove_database": "1", "remove_database_user": "1"} }'

Where:

- application_id - the ID of an application
- remove_database_user - set 1 to remove database user, otherwise set 0
**remove_database** - set 1 to remove database, otherwise set 0
**remove_directory** - set 1 to remove directory, otherwise set 0
**remove_data_directory** - set 1 to remove data directory, otherwise set 0

### 12.7 Get List of All Application Backups

To get the list of all backups taken for this particular application server, use the following request::

GET
/application_servers/:application_server_id/applications/backups.xml
GET
/application_servers/:application_server_id/applications/backups.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<backups type="array">
  <backup>
    <application_id>123_70977</application_id>
    <application_type>CodeIgniter</application_type>
    <backup_note>Code Igniter Backup #1</backup_note>
    <identifier>475d8fd0e008941b19c9819d8dc8a410</identifier>
    <software_version>3.0.0</software_version>
    <size>1.369 MB</size>
  </backup>
  <backup>
    <application_id>123_70977</application_id>
    <application_type>CodeIgniter</application_type>
    <backup_note>Code Igniter Backup #1</backup_note>
    <identifier>7d0c9305d8e816282e17e432903e33eb</identifier>
    <software_version>3.0.0</software_version>
    <size>1.369 MB</size>
  </backup>
  ...
</backups>
```

Where:

- **application_id** - the ID of the application, which was backed up
- **application_type** - the name of the application, which was backed up
backup_note - the note, that was written during backup creation

identifier - identifier of the application backup

software_url - this URL is a link to application itself

software_version - the version of application software

size - backup size

12.8 Restore Application Backup

To restore application backup, use the following request:

POST /application_servers/:application_server_id/applications/backups/:identifier/restore.xml

POST /application_servers/:application_server_id/applications/backups/:identifier/restore.json

XML Request Example


JSON Request Example


Where:

restore_directory - set 1 to restore directory, otherwise set 0

restore_database - set 1 to restore database, otherwise set 0

identifier - identifier of the application backup. It can be found using the Get List of All Application Backups request.

12.9 Remove Application Backup

To remove application backup, use the following request:

DELETE /application_servers/:application_server_id/applications/backups/:identifier/destroy.xml
DELETE
/application_servers/:application_server_id/applications/backups/:identifier/destroy.json

XML Request Example

curl -i -X DELETE -u user:userpass --url
http://onapp.test/application_servers/2/applications/backups/tfgguk6iyu/destroy.xml
-H 'Content-type: application/xml' -H 'Accept: application/xml'

JSON Request Example

curl -i -X DELETE -u user:userpass --url
http://onapp.test/application_servers/2/applications/backups/tfgguk6iyu/destroy.json
-H 'Content-type: application/json' -H 'Accept: application/json'

Where:
identifier - identifier of the application backup. It can be found using the Get List of All Application Backups request.

12.10 System Applications

There are system applications within an application server. You can install or switch PHP versions by means of system applications.

12.10.1 Get List of System Applications

To get the list of system applications, use the following request:
GET /application_servers/:application_server_id/system_apps.xml
GET /application_servers/:application_server_id/system_apps.json

XML Request Example

curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url
http://onapp.test/application_servers/2/system_apps.xml

JSON Request Example

curl -i -X GET -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' --url
http://onapp.test/application_servers/2/system_apps.json

XML Output Example
<system_apps type="array">
 <system_app>
  <api_name>php54</api_name>
  <default type="boolean">true</default>
  <id>2</id>
  <installed type="boolean">true</installed>
  <name>PHP 5.4</name>
  <version>5.4.45</version>
 </system_app>
</system_apps>

Where:

*api_name* - the API name of a system application

*id* - the ID of a system application

*name* - the name of a system application

*version* - the version number of a system application

### 12.10.2 Install System Application

To install a system application, use the following request:

**PUT**

```
/application_servers/:application_server_id/system_apps/system_app_id/install.xml
```

**PUT**

```
/application_servers/:application_server_id/system_apps/system_app_id/install.json
```

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/6/system_apps/143/install.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/6/system_apps/143/install.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

*application_server_id* - the ID of an application server where you want to install a system application

*system_app_id* - the ID of the system application which you want to install

You can get system application ID with the [Get List of System Applications](#) API request.
12.10.3 Switch PHP Version

To switch a PHP version, use the following request:

```
PUT /application_servers/:id/settings/switch_php_version.xml
PUT /application_servers/:id/settings/switch_php_version.json
```

**XML Request Example**

```
curl -i -X PUT -u user:userpass
'Accept: application/xml' -H 'Content-type: application/xml' -d
'<?xml version="1.0"?><php_version>php55</php_version>'
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass
'Accept: application/json' -H 'Content-type: application/json' -d
'{"php_version":"php55"}'
```

Where:

*php_version* - the name of PHP version to which you want to switch

12.10.4 Uninstall System Application

To uninstall a system application, use the following request:

```
PUT /application_servers/:application_server_id/system_apps/system_app_id/uninstall.xml
PUT /application_servers/:application_server_id/system_apps/system_app_id/uninstall.json
```

**XML Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/6/system_apps/88/uninstall.xml -H
'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/6/system_apps/88/uninstall.json -H
'Accept: application/json' -H 'Content-type: application/json'
```

Where:

*application_server_id* - the ID of an application server where you want to uninstall a system application

*system_app_id* - the ID of the system application which you want to uninstall
12.11 Domains

This section contains the API requests you can apply to manage Application Server domains.

12.11.1 Get List of Domains

To get the list of application server domains, use the following request:

GET /application_servers/:application_server_id/domains.xml
GET /application_servers/:application_server_id/domains.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```xml
<domains type="array">
  <domain>
    <identifier>3dd0d7734983e2db8ba7677bd5b1la70</identifier>
    <name>domain_name</name>
    <path>/home/onapp/public_html</path>
    <type>primary</type>
  </domain>
  ...
</domains>
```

Where:
- **identifier** - the domain's identifier
- **name** - the name of the domain
- **path** - the route to the domain folder
- **type** - the type of the domain

12.11.2 Create Domain

To create a domain, use the following request:
There are several options for domain creation:

- **Addon domain creation with custom path**
- **Domain addition to an existing application**
- **Creation of parked domain**

### 12.11.2.1 Addon domain creation with custom path

**XML Request Example**

```bash
curl -i -X POST -u user:us
erverpass
http://onapp.test/application_servers/6/domains.xml
-H 'Accept: application/xml'
-H 'Content-Type: application/xml'
-d '<domain><path>ZikulaTest</path><domain>addon.com</domain></domain>'
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/6/domains.json
-H 'Accept: application/json'
-H 'Content-Type: application/json'
-d '{"domain": {"path": "ZikulaTest","domain": "addon.com"}}'
```

**Where:**

- **domain** - specify the name of the domain
- **path** - indicate the route to domain folder

### 12.11.2.2 Domain addition to an existing application

**XML Request Example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/6/domains.xml
-H 'Accept: application/xml'
-H 'Content-Type: application/xml'
-d '<domain><domain>existed.com</domain><application_id>None</application_id></domain>'
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/6/domains.json
-H 'Accept: application/json'
-H 'Content-Type: application/json'
-d '{"domain": {"domain": "existed.com","application_id": null}}'
```

**Where:**

- **domain** - the name of the domain
- **application_id** - the ID of an application to which you want to add the domain
12.11.2.3 Creation of parked domain

XML Request Example

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/6/domains.xml
'Accept: application/xml' -H 'Content-type: application/xml' -d
'<domain><domain>parked.com</domain><application_id>Null</application_id></domain>'
```

JSON Request Example

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/6/domains.json
'Accept: application/json' -H 'Content-type: application/json' -d
{"domain": "parked.com", "application_id": null}
```

Where:

domain - the name of the domain

12.11.3 Delete Domain

To delete a domain, use the following request:

DELETE
/application_servers/:application_server_id/domains/:domain_identifier.xml

DELETE
/application_servers/:application_server_id/domains/:domain_identifier.json

XML Request Example

```bash
curl -i -X DELETE -u user:userpass
http://onapp.test/application_servers/109/domains/03e7a0bf970e4da3.xml
```

JSON Request Example

```bash
curl -i -X DELETE -u user:userpass
http://onapp.test/application_servers/109/domains/03e7a0bf970e4da3.json
```

Where:

domain_identifier - the identifier of the domain you want to delete.

You can check identifier of the required domain with GET method:

GET /application_servers/:application_server_id/domains.xml

GET /application_servers/:application_server_id/domains.json
12.12 FTP Users

This section contains the API requests you can apply to manage FTP users.

12.12.1 Get List of FTP Users

To get the list of FTP users, use the following request:

GET /application_servers/:application_server_id/ftp_users.xml
GET /application_servers/:application_server_id/ftp_users.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<ftp_users type="array">
    <ftp_user>
        <identifier>eb9baa2f0c5043e4535a17862e5f2ab5</identifier>
        <login>onapp</login>
        <path>/home/onapp</path>
        <usage type="integer">0</usage>
    </ftp_user>
    <ftp_user>
        <identifier>0fd3345dcc4b4c20b46f8ee33f52ba75</identifier>
        <login>test_onapp.test</login>
        <path>/home/onapp/www/test</path>
        <usage type="integer">0</usage>
    </ftp_user>
    ...
</ftp_users>
```

Where:

- **identifier** - the user's identifier
- **login** - the user's login name
- **path** - the route to FTP folder
- **usage** - the amount of FTP folder space, used by this user (in MB)
12.12.2 Create FTP User

To create an FTP user, use the following request:

```
POST /application_servers/:application_server_id/ftp_users.xml
POST /application_servers/:application_server_id/ftp_users.json
```

**XML Request Example**

```
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/ftp_users.xml
'Accept: application/xml' -H 'Content-type: application/xml' -d
'<!ftp_user><password>1234</password><password_confirmation>1234</password_confirmation><login>login364</login><path>www/usr2</path></ftp_user>'
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/ftp_users.json
'Accept: application/json' -H 'Content-type: application/json' -d
'{ "ftp_user": { "password": "1234", "password_confirmation": "1234", "login": "login364", "path": "www/usr2" } }'
```

Where:
- `password` - create user's password
- `password_confirmation` - enter user's password one more time
- `login` - provide user's login name
- `path` - indicate the route to FTP folder

12.12.3 Delete FTP User

To delete an FTP user, use the following request:

```
DELETE /application_servers/:application_server_id/ftp_users/:ftp_user_id_identifier.xml
DELETE /application_servers/:application_server_id/ftp_users/:ftp_user_id_identifier.json
```

**XML Request Example**

```
curl -i -X DELETE -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/application_servers/109/ftp_users/hj09-jhkfd09.xml
```

**JSON Request Example**

```
curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/application_servers/109/ftp_users/hj09-jhkfd09.json
```
Where:

ftp_user_identifier - the identifier of the user you want to delete.

You can check identifier of the required FTP user with GET method:

GET /application_servers/:application_server_id/ftp_users.xml
GET /application_servers/:application_server_id/ftp_users.json

12.13 Databases

This section contains the API requests you can apply to manage databases available for your Application Server.

12.13.1 Get List of Databases

To get the list of databases, use the following request:

GET /application_servers/:application_server_id/databases.xml
GET /application_servers/:application_server_id/databases.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```
<databases type="array">
  <database>
    <db>Test1</db>
  </database>
  <database>
    <db>cmfsfs</db>
  </database>
</databases>
```

Where:

db - the name of the database
12.13.2  Create Database

To create a database, use the following request:

POST /application_servers/:application_server_id/databases.xml
POST /application_servers/:application_server_id/databases.json

**XML Request Example**

curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/databases.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'<database><db>Test1</db></database>'

**JSON Request Example**

curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/databases.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d '{"database": {"db":"Test1"}}'

Where:

*db* - the name of the database

12.13.3  Delete Database

To delete a database, use the following request:

DELETE /application_servers/:application_server_id/databases/:db.xml
DELETE /application_servers/:application_server_id/databases/:db.json

**XML Request Example**

curl -i -X DELETE -u user:userpass
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d
-url 'http://onapp.test/application_servers/109/databases/database_1.xml'

**JSON Request Example**

curl -i -X DELETE -u user:userpass
-H 'Accept: application/json' -H 'Content-type: application/json' -d
-url 'http://onapp.test/application_servers/109/databases/database_1.json'

Where:

*db* - the name of the database you want to delete.

You can check name of the required database with GET method:

GET /application_servers/:application_server_id/databases.xml
GET /application_servers/:application_server_id/databases.json
12.13.4 Get List of Database Users

To get the list of database users, use the following request:
GET /application_servers/:application_server_id/database_users.xml
GET /application_servers/:application_server_id/database_users.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<database_users type="array">
  <database_user>
    <name>user</name>
  </database_user>
  <database_user>
    <name>cmfsfs</name>
  </database_user>
</database_users>
```

Where:

- **name**: the name of the database user

12.13.5 Get List of Users Assigned to Database

To get the list of users assigned to a certain database and their permissions, use the following request:
GET /application_servers/:application_server_id/databases/:db/privileges.xml
GET /application_servers/:application_server_id/databases/:db/privileges.json

**XML Request Example**

JSON Request Example


Where:

db - the name of the required database

XML Output Example

```
<database_users type="array">
  <database_user>
    <prilist>
      <DELETE>false</DELETE>
      <SHOW_VIEW>false</SHOW_VIEW>
      <HOST=localhost</HOST>
      <DROP>false</DROP>
      <CREATE>false</CREATE>
      <CREATE_VIEW>false</CREATE_VIEW>
      <EXECUTE>false</EXECUTE>
      <TRIGGER>false</TRIGGER>
      <INDEX>false</INDEX>
      <CREATE_TEMPORARY_TABLES>false</CREATE_TEMPORARY_TABLES>
      <LOCK_TABLES>false</LOCK_TABLES>
      <REFERENCES>false</REFERENCES>
      <SELECT>true</SELECT>
      <INSERT>false</INSERT>
      <UPDATE>false</UPDATE>
      <ALTER>false</ALTER>
    </prilist>
    <name>test_db</name>
  </database_user>
</database_users>
```

Where:

prilist - the list of privileges (permissions) assigned to the database user.

Below you can find the list of privileges:

- SELECT
- CREATE
- INSERT
- UPDATE
- ALTER
- DELETE
- INDEX
- CREATE_TEMPORARY_TABLES
- EXECUTE
- DROP
12.13.6 Create Database User

To create a database user, use the following request:

POST /application_servers/:application_server_id/database_users.xml
POST /application_servers/:application_server_id/database_users.json

XML Request Example

```
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/database_users.xml
-H 'Accept: application/xml' -H 'Content-Type: application/xml'
-d '<database_user><name>user</name><password>pass</password></database_user>
'
```

JSON Request Example

```
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/database_users.json
-H 'Accept: application/json' -H 'Content-Type: application/json'
-d '{"database_user": {"name": "user", "password": "pass"}}'
```

Where:

name - the name of the database user

The length of the name should not exceed 11 characters.

password - the password for the database user

12.13.7 Assign User to Database

To assign user to a database, use the following request:

POST /application_servers/:application_server_id/databases/:db/assign_user.xml
POST /application_servers/:application_server_id/databases/:db/assign_user.json

XML Request Example

```
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/databases/10/database_users.xml
-H 'Accept: application/xml' -H 'Content-Type: application/xml'
-d '<database_user><name>user</name><password>pass</password></database_user>
'
```
OnApp Cloud 6.3 API Guide

**JSON Request Example**

```bash
```

Where:

- **db** - the name of the database to which you want to assign users
- **db_user** - the name of the database user
- **prilist** - the list of privileges (permissions) which you want to assign to the database user. Set 'true' to the specific privilege to assign it to the user.

Below you can find the list of privileges:

- SELECT
- CREATE
- INSERT
- UPDATE
- ALTER
- DELETE
- INDEX
- CREATE_TEMPORARY_TABLES
- EXECUTE
- DROP
- LOCK_TABLES
- REFERENCES
- CREATE_ROUTINE
- CREATE_VIEW
- SHOW_VIEW
- TRIGGER

- **host** - indicate a host name for the database. There are several options of host names:
  - localhost - specify this parameter to choose a local host name
  - % - specify this parameter to choose any host name
  - 27.0.0.1 - specify any IP Address

### 12.13.8 Update Database User Privileges

To update a database user privileges, use the following request:

```bash
PUT /application_servers/:application_server_id/database_users/:name/privileges.xml
```

```bash
PUT /application_servers/:application_server_id/database_users/:name/privileges.json
```
XML Request Example

curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/109/database_users/name_one/privileges.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d ' 
<database_user><db_name>label</db_name><host>localhost</host><prilist><SELECT>true</SELECT></prilist></database_user> 

JSON Request Example

curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/109/database_users/name_one/privileges.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d ' 
{"database_user": {"db_name": "label", "host": "localhost", "prilist":{"SELECT":true}}}} 

Where:

**db_name** - the name of the database

**prilist** - the list of privileges (permissions) which you want to assign to the database user. Put 'true' next to the specific privilege to assign it to the user.

Below you can find the list of privileges:

- SELECT
- CREATE
- INSERT
- UPDATE
- DELETE
- INDEX
- CREATE_TEMPORARY_TABLES
- EXECUTE
- DROP
- LOCK_TABLES
- REFERENCES
- CREATE_ROUTINE
- CREATE_VIEW
- SHOW_VIEW
- TRIGGER

**host** - indicate a host name for the database. There are several options of host names:

- localhost - specify this parameter to choose a local host name
- % - specify this parameter to choose any host name
- 27.0.0.1 - specify any IP Address

---

12.13.9 Change Database User Password

To change a database user password, use the following request:

PUT
/application_servers/:application_server_id/database_users/:name/change_password.xml
PUT
/application_servers/:application_server_id/database_users/:name/change_password.json

XML Request Example

curl -i -X PUT -u user:userpass

JSON Request Example

curl -i -X PUT -u user:userpass

Where:
password - the password for the database user

12.13.10 Unassign User from Database

To unassign user from a database, use the following request:

PUT
/application_servers/:application_server_id/database_users/:name/privileges.xml

PUT
/application_servers/:application_server_id/database_users/:name/privileges.json

XML Request Example

curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/109/database_users/name_one/privileges.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<database_user><db_name>label</db_name></database_user>'

JSON Request Example

curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/109/database_users/name_one/privileges.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"database_user": {"db_name": "label"}}'

Where:

db_name - the name of the database

12.13.11 Delete Database User

To delete a database user, use the following request:
OnApp Cloud 6.3 API Guide

DELETE /application_servers/:application_server_id/database_users/:name.xml
DELETE /application_servers/:application_server_id/database_users/:name.json

**XML Request Example**
```
```

**JSON Request Example**
```
```

**Where:**

`name` - the name of the database user you want to delete.

You can check name of the required database user with GET method:

GET /application_servers/:application_server_id/database_users.xml
GET /application_servers/:application_server_id/database_users.json

### 12.14 Email Accounts

This section contains the API requests you can apply to manage email accounts for domains.

#### 12.14.1 Get List of Email Accounts

To get the list of email accounts, use the following request:

GET /application_servers/:application_server_id/email_accounts.xml
GET /application_servers/:application_server_id/email_accounts.json

**XML Request Example**
```
```

**JSON Request Example**
To get the list of email accounts for a specific domain, use the following request:

GET
/application_servers/:application_server_id/email_accounts?domain=example.com.xml

GET
/application_servers/:application_server_id/email_accounts?domain=example.com.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```xml
<email_accounts type="array">
  <email_account>
    <count type="integer">0</count>
    <identifier>27b5fcd7f638f8088ealc78196c37ae9</identifier>
    <user>test@109.123.91.19</user>
  </email_account>
  <email_account>
    <count type="integer">0</count>
    <identifier>8fbbca4101c61b8c5d0157353dc8b</identifier>
    <user>test2@109.123.91.19</user>
  </email_account>
</email_accounts>
```

**Where:**

- **identifier** - the email account's identifier
- **user** - the email account
Where:

*identifier* - the email account's identifier

*user* - the email account

### 12.14.3 Create Email Account

To create an email account, use the following request:

POST `/application_servers/:application_server_id/email_accounts.xml`

POST `/application_servers/:application_server_id/email_accounts.json`

**XML Request Example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/email_accounts.xml
'Accept: application/xml' -H 'Content-type: application/xml' -d
'<?xml version="1.0" encoding="UTF-8"?>
<email_accounts type="array">
  <email_account>
    <count type="integer">0</count>
    <identifier>27b5fcd7f638f8088ea1c78196c37ae9</identifier>
    <space></space>
    <user>test@109.123.91.19</user>
  </email_account>
  <email_account>
    <count type="integer">0</count>
    <identifier>8fbbcfa4101c61b5b8c5d0157353dc8b</identifier>
    <space></space>
    <user>test2@109.123.91.19</user>
  </email_account>
</email_accounts>
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/109/email_accounts.json
'Accept: application/json' -H 'Content-type: application/json' -d
'{"email_account": {"password": "1234", "password_confirmation": "1234", "user": "login364", "domain": "example.com"}}'
```

Where:

*password* - create a password for this email account

*password_confirmation* - repeat the password to confirm it

*user* - add text, which will be the part of email account before the @ symbol

*domain* - indicate the specific domain

### 12.14.4 Delete Email Account for Default Domain

To delete an email account, use the following request:
DELETE
/application_servers/:application_server_id/email_accounts/:email_account_identifier.xml

DELETE
/application_servers/:application_server_id/email_accounts/:email_account_identifier.json

**XML Request Example**


**JSON Request Example**


**Where:**

*email_account_identifier* - the identifier of the email account you want to delete.

You can check identifier of the required email account with GET method:

GET /application_servers/:application_server_id/email_accounts.xml
GET /application_servers/:application_server_id/email_accounts.json

---

12.14.5 **Delete Email Account for Specific Domain**

To delete an email account for a specific domain, use the following request:

DELETE
/application_servers/:application_server_id/email_accounts/:email_account_identifier.xml

DELETE
/application_servers/:application_server_id/email_accounts/:email_account_identifier.json

**XML Request Example**


**JSON Request Example**

Where:

email_account_identifier - the identifier of the email account you want to delete.
domain_name - the name of a specific domain

You can check identifier of the required email account with GET method:
GET /application_servers/:application_server_id/email_accounts.xml
GET /application_servers/:application_server_id/email_accounts.json

12.15 Services

This section contains the API requests you can apply to manage application server services.

12.15.1 Get List of Services

To get the list of services, use the following request:
GET /application_servers/:application_server_id/services.xml
GET /application_servers/:application_server_id/services.json

XML Request Example


JSON Request Example


XML Output Example
<services type="array">
  <service>
    <id>16</id>
    <name>MySQL</name>
    <status>stop</status>
  </service>
  <service>
    <id>3</id>
    <name>Apache</name>
    <status>running</status>
  </service>
  <service>
    <id>34</id>
    <name>BIND</name>
    <status>running</status>
  </service>
</services>

Where:

- **id** - the ID of the service
- **name** - the service name
- **status** - the service status

### 12.15.2 Start/Stop/Rerstart Services

You can fulfill the following actions with services:

- **start service**
- **stop service**
- **restart service**

You can check ID of the required service with GET method:

GET /application_servers/:application_server_id/services.xml
GET /application_servers/:application_server_id/services.json

#### 12.15.2.1 Start service

To start service, use the following request:

PUT /application_servers/:application_server_id/services/:service_id/start.xml
PUT /application_servers/:application_server_id/services/:service_id/start.json

**XML Request Example**

```
```

**JSON Request Example**

...
12.15.2.2 Stop service
To stop service, use the following request:

PUT
/application_servers/:application_server_id/services/:service_id/stop.
xml
PUT
/application_servers/:application_server_id/services/:service_id/stop.
json

XML Request Example

```
```

JSON Request Example

```
```

12.15.2.3 Restart service
To restart service, use the following request:

PUT
/application_servers/:application_server_id/services/:service_id/restart.
xml
PUT
/application_servers/:application_server_id/services/:service_id/restart.
json

XML Request Example

```
```

JSON Request Example

```
```
13 Assets

Assets are the compute resources that are connected to the control panel server but are either not configured or not assigned to the compute zone. You can retrieve the list of assets using the Get List of Unassigned Assets request.

13.1 Get List of Assets

To view the list of assets, use the following request:

GET settings/assets.xml
GET settings/assets.json

XML Request Example


JSON Request Example


XML Output Example

<assets type="array">
  <asset>
    <mac>00:15:17:63:ea:3e</mac>
  </asset>
  <asset>
    <mac>00:15:17:5a:51:4a</mac>
  </asset>
</assets>

Where:

mac - asset MAC address

To view the list of assets that are already created but not assigned to the compute zone, use the Get List of Unassigned Assets request.

13.2 Get Asset Details

To view the list of assets, use the following request:

GET /settings/assets/:asset_mac_address.xml
GET /settings/assets/:asset_mac_address.json

XML Request Example
curl -i -X GET -u user:userpass

**JSON Request Example**

curl -i -X GET -u user:userpass

**XML Response Example**

```xml
<asset>
  <mac nil="true"/>
  <disks type="array">
    <disk>
      <name>disk1name1</name>
      <scsi>disk1scsi</scsi>
    </disk>
    <disk>
      <name>disk3name</name>
      <scsi>disk3scsi</scsi>
    </disk>
  </disks>
  <nics type="array">
    <nic>
      <name>nic1name</name>
      <mac>nic1mac</mac>
    </nic>
    <nic>
      <name>eth1</name>
      <mac>00:1a:64:62:41:e1</mac>
    </nic>
  </nics>
  <pcis type="array">
    <pci>
      <name>Broadcom Corporation NetXtreme BCM5704 Gigabit Ethernet [14e4:1648] (rev 10)</name>
      <pci>02:01:0</pci>
    </pci>
  </pcis>
</asset>
```

Where:

- **mac** - asset MAC address
- **disk** - an array of asset's disks along with the following details:
  - **name** - disk label
  - **scsi** - SCSI inquiry product revision number
- **nics** - an array of asset's network interfaces along with their details:
  - **name** - NIC name
  - **mac** - NIC MAC address
- **pcis** - an array of NIC PCIs along with the following details:
name - PCI label
pci - NIC PCI

13.3 Get List of Unassigned Assets

To view the list of assets, use the following request:
GET /hypervisors/not_grouped.xml
GET /hypervisors/not_grouped.json

XML Request Example

curl -i -X GET -u user:userpass
http://onapp.test/hypervisors/not_grouped.xml -H 'Accept: application/xml'
- H 'Content-type: application/xml'

JSON Request Example

curl -i -X GET -u user:userpass
http://onapp.test/hypervisors/not_grouped.json -H 'Accept: application/json'
- H 'Content-type: application/json'

XML Output Example
Where:

**backup** - true, if the CloudBoot compute resource is used as a backup server. This parameter is for CloudBoot compute resources only. For other compute resource types the **backup** value is 0.
backup_ip_address - provisioning network IP address

blocked - true if the compute resource is blocked, otherwise false

built - true if the compute resource is built, otherwise false

called_in_at – the date when the compute resource was called in the [YYYY][MM][DD][hh][mm][ss]Z format

collection_options - an array of the following vCenter cluster parameters:

- login - vCenter login
- password - vCenter password
- cluster_name - vCenter cluster name
- distributed_virtual_switch_name - distributed virtual switch label

cpu_idle - time of CPU delay

cpu_mhz - CPU operating frequency

created_at – the date in the [YYYY][MM][DD][hh][mm][ss]Z format

custom_config - custom commands that run when the compute resource is booted

disable_failover – true if compute resource failover is disabled, otherwise false.

disk_pcis - comma-separated list of hypervisor disk pcis

distro - distributive label

enabled - true, if the compute resource is enabled, otherwise false

failure_count – the number of failures

format_disks - true, if the compute resource's disks are formatted during creation, otherwise false

free_mem - free compute resource memory

host - host label

hypervisor_group_id - the ID of a compute zone to which this compute resource is attached

hypervisor_type - the compute resource type

id - the compute resource ID

ip_address – the compute resource IP address

label - the compute resource label

list_of_logical_volumes - an array of compute resource logical volumes

list_of_volume_groups - an array of compute resource volume groups

list_of_zombie_domains - an array of zombie virtual servers

locked - true if the compute resource is locked, otherwise false
mac - compute resource MAC address

machine - architecture type

mem_info - Xen compute resource Dom0 size. This parameter is for Xen compute resources only. For other compute resource types the mem_info value is 0.

mtu - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.

The maximum transportation unit (MTU) is the maximum size of a unit that can be transmitted transferred via ethernet traffic. Any data that exceed the specified MTU value will be divided into smaller units before being transferred. Utilization of jumbo frames allows to reduce increase throughput and increase CPU utilization during large size file transfers.

online - true if online, otherwise false

ovs - whether the OpenvSwitch is enabled or not. Redundant parameter.

passthrough_disks - true, if the disks are passed through to the storage nodes, otherwise 0

release - compute resource kernel version

server_type - server type:
- virtual
- smart
- baremetal

spare – true if no VSs are assigned, otherwise false

storage_channel - storage channel for the communication

threads_per_core - compute resource core and CPU ratio. For example, if threads_per_core = 2, compute resource with 2 cores will have 4 CPUs.

total_mem - total compute resource memory

total_zombie_mem - memory space occupied by zombie disks

updated_at – the date when the record was made in the DB in the [YYYY][MM][DD][hh][mm][ss]Z format

uptime - compute resource uptime value; shows how long the compute resource is online

vmware_total_cpu_cores - the total number of VMware compute resource CPU cores

total_cpus – the number of virtual cores

free_memory – free RAM (MB) of compute resource

used_cpu_resources – the percentage of used CPU resources
**total_memory** – total RAM (MB) of compute resource

**cpu_cores** – the number physical of cores per compute resource

**free_disk_space** - free compute resource disk space in GB

**memory_allocated_by_running_vms** - the compute resource RAM in MB allocated to the virtual servers, which are currently running on this compute resource

**total_memory_allocated_by_vms** - the compute resource RAM in MB allocated to all virtual servers of this compute resource
14 Auto-Backups

OnApp Cloud provides a range of auto-backup possibilities for Virtual Servers:

- See Auto-Backup Presets to learn how to change the auto-backup schedule, which applies during the VS creation, or when the auto-backup is enabled for the first time.
- See Manage Auto-Backups chapter to learn how to enable or disable auto-backups for already existing Virtual Servers.
- See Schedules to learn how to view, create, delete or change any schedule for a particular Virtual Server.

14.1 Auto-backup Presets

Auto-backup presets are simple way to set up an automatic backup schedule when virtual servers are created or when the auto-backup is enabled for the first time. Once configured, they can be applied to a virtual server automatically when the Automatic backups required parameter is enabled during VS creation.

14.1.1 Get List of Auto-backup Presets

To get the list of available auto-backup presets, use the following request:

GET /settings/autobackup_presets.xml
GET /settings/autobackup_presets.json

XML Request Example

```
```

JSON Request Example

```
```

An array of auto-backup presets is returned. If there are no presets, an empty array is returned.

XML Output Example
<autobackup_templates type="array">
    <autobackup_template>
        <id type="integer">1</id>
        <enabled type="boolean">true</enabled>
        <duration type="integer">1</duration>
        <period>days</period>
        <created_at type="datetime">2018-07-14T15:01:38Z</created_at>
        <updated_at type="datetime">2018-07-28T11:49:52Z</updated_at>
        <rotation_period type="integer">1</rotation_period>
    </autobackup_template>
    <autobackup_template>
        <id type="integer">2</id>
        <enabled type="boolean">true</enabled>
        <duration type="integer">1</duration>
        <period>weeks</period>
        <created_at type="datetime">2018-07-14T15:01:38Z</created_at>
        <updated_at type="datetime">2018-07-28T11:50:21Z</updated_at>
        <rotation_period type="integer">1</rotation_period>
    </autobackup_template>
    <autobackup_template>...</autobackup_template>
</autobackup_templates>

Where:

- **id** - the ID of the auto-backup preset
- **enabled** - true if the auto-backup preset is enabled, otherwise, false
- **duration** - the number specifying how often a backup should be taken
- **period** - the time period (days, weeks, months, or years)
- **created at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **updated at** - the date when the auto-backup preset was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **rotation_period** - the number of backups to keep before deleting the initial backup

### 14.1.2 Get Auto-backup Preset Details

To get the details of a particular auto-backup preset, use the following request:

**GET /settings/autobackup_presets/:id.xml**

**GET /settings/autobackup_presets/:id.json**

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```
XML Output Example

```
<autobackup_templates type="array">
  <autobackup_template>
    <id type="integer">1</id>
    <enabled type="boolean">true</enabled>
    <duration type="integer">1</duration>
    <period>days</period>
    <created_at type="datetime">2018-07-14T15:01:38Z</created_at>
    <updated_at type="datetime">2018-07-28T11:49:52Z</updated_at>
    <rotation_period type="integer">1</rotation_period>
  </autobackup_template>
</autobackup_templates>
```

Where:
- **id** - the ID of the auto-backup preset
- **enabled** - true if the auto-backup preset is enabled, otherwise, false
- **duration** - the number specifying how often a backup should be taken
- **period** - the time period (days, weeks, months, or years)
- **created at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated at** - the date when the auto-backup preset was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **rotation_period** - the number of backups to keep before deleting the initial backup

14.1.3 Edit Auto-backup Preset

To edit an auto-backup preset, use the following request:

**XML Request Example**
```
```

**JSON Request Example**
```
```

Where:
- **duration** - edit the number specifying how often a backup should be taken
- **enabled** - set true if auto-backup preset is enabled, otherwise set false
- **rotation_period** - the number of backups to keep before deleting the initial backup
Every *autobackup_preset_id* has its defined period (either days, or weeks, or months, or years), which cannot be edited.

Returns HTTP 204 response on success, or HTTP 404 when an auto-backup preset with the ID specified is not found, or the URL requested is incorrect.

### 14.2 Manage Auto-Backups

Auto-backup has a specific target from which the backup is taken according to the schedule assigned to such target. When you use a normal backup scheme in your cloud - the target for auto-backups will be a disk of the Virtual Server; for incremental backups - it will be the whole Virtual Server.

Windows-based Virtual Servers support only normal backups. Thus, even if incremental backups are selected for your cloud, the auto-backup target for Windows VS will be its disk.

The auto-backup may be enabled either during the creation of Virtual Server using the `required_automatic_backup` parameter, or at any time afterwards. If you enable the auto-backup for a Virtual Server or its disk for the first time (i.e. there is no auto-backup schedule for such target) the system will automatically apply the schedule configured in auto-backup presets.

If you enable the auto-backups for a target with assigned schedule - this schedule will remain unchanged.

#### 14.2.1 Enable Auto-backups for VS

To enable incremental auto-backups for a virtual server, use the following request:

```bash
POST /virtual_machines/:id/autobackup_enable.xml
POST /virtual_machines/:id/autobackup_enable.json
```

**XML Request Example**

```bash
curl -i -X POST http://onapp.test/virtual_machines/2/autobackup_enable.xml
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
```

#### 14.2.2 Disable Auto-backups for VS

To disable incremental auto-backups for a virtual server, use the following request:
14.2.3 Enable Auto-backups for Disk

To enable auto-backups for a disk, use the following request:

POST /settings/disks/:disk_id/autobackup_enable.xml
POST /settings/disks/:disk_id/autobackup_enable.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where:

disk_id* - is the ID of the disk, for which you want to enable auto-backup

14.2.4 Disable Auto-backups for Disk

To disable auto-backups for a disk, use the following request:

POST /settings/disks/:id/autobackup_disable.xml
POST /settings/disks/:id/autobackup_disable.json

**XML Request Example**

```
```

**JSON Request Example**

```
```
JSON Request Example

```
curl -i -X POST http://onapp.test/settings/disks/4/autobackup_disable.json
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:
*disk_id*- the ID of the disk for which you want to disable auto-backup

14.3 Schedules

Schedules are concerned with backups scheduled for virtual servers or virtual server disks in the cloud:

- If you are using normal backups, the schedules are created per disk.
- If you are using incremental backups, the schedules are created per virtual server.

When a schedule is no longer needed, it can be deleted so that the task will no longer run.

14.3.1 Get List of All Schedules

To get the list of all schedules, use the following request:

GET /settings/schedules.xml
GET /settings/schedules.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<schedules>
<schedule>
  <action>autobackup</action>
  <created_at type="datetime">2013-12-03T17:38:24+03:00</created_at>
  <duration type="integer">1</duration>
  <failure_count type="integer">0</failure_count>
  <id type="integer">131</id>
  <period type="months"/>
  <rotation_period type="integer">1</rotation_period>
  <start_at type="datetime">2014-01-03T17:38:24+03:00</start_at>
  <status>enabled</status>
  <target_id type="integer">11542</target_id>
  <target_type>Disk</target_type>
  <updated_at type="datetime">2013-12-03T17:38:37+03:00</updated_at>
  <user_id type="integer">1875</user_id>
  <schedule_logs type="array">
    <schedule_log>
      <created_at type="datetime">2013-12-03T17:38:37+03:00</created_at>
      <id type="integer">415</id>
      <log_output>Executing Rollback...</log_output>
      <schedule_id type="integer">131</schedule_id>
      <status>failed</status>
      <updated_at type="datetime">2013-12-03T17:38:37+03:00</updated_at>
    </schedule_log>
  </schedule_logs>
</schedule>
</schedules>

Where:

- **action** – backup type
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** - the date when a schedule was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **start_at** - the date when a backup started in the [YYYY][MM][DD][hh][mm][ss]Z format
- **duration** - how often a backup is taken
- **failure_count** - the number of requests processed until the task fails
- **id** - the schedule ID
- **period** - time period for a backup schedule (days, weeks, months, or years)
- **rotation_period** - the number of backups after which the first backup will be deleted
- **target_id** - the disk ID for which a backup is taken
- **target_type** - currently, you can schedule backup of Disks only
- **user_id** - the ID of a user who created this schedule
- **failure_count** - the number of requests processed until the task fails
- **status** - the status of the backup schedule (enabled, disabled, or failed)
- **schedule_logs** – schedule’s log messages with the following details:
  - **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
  - **updated_at** - the date when in the [YYYY][MM][DD][hh][mm][ss]Z format
  - **id** – log message ID
  - **log_output** – log message text
- `schedule_id` – ID of a schedule to which the log message belongs to
- `status` – status of a scheduled backup performed

**Page History**

v.3.2
- added `rotation_period` parameter

### 14.3.2 Get Schedule Details

To get details for a particular disk backup schedule, use the following request:

GET /settings/schedules/:id.xml
GET /settings/schedules/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

This method outputs an array of the disk backups scheduled within your cloud. If there are no schedules, an empty array is returned.

**XML Output Example**
<schedule>
  <action>autobackup</action>
  <created_at type="datetime">2013-12-03T17:38:24+03:00</created_at>
  <duration type="integer">1</duration>
  <failure_count type="integer">0</failure_count>
  <id type="integer">131</id>
  <params nil="true"/>
  <period>months</period>
  <rotation_period type="integer">1</rotation_period>
  <start_at type="datetime">2014-01-03T17:38:24+03:00</start_at>
  <status>enabled</status>
  <target_id type="integer">11542</target_id>
  <target_type>Disk</target_type>
  <updated_at type="datetime">2013-12-03T17:38:37+03:00</updated_at>
  <user_id type="integer">1875</user_id>
  <schedule_logs type="array">
    <schedule_log>
      <created_at type="datetime">2013-12-03T17:38:37+03:00</created_at>
      <id type="integer">415</id>
      <log_output>Executing Rollback...
      Fatal: []</log_output>
      <schedule_id type="integer">131</schedule_id>
      <status>failed</status>
      <updated_at type="datetime">2013-12-03T17:38:37+03:00</updated_at>
    </schedule_log>
  </schedule_logs>
</schedule>

Where:

**action** – backup type

**created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**updated_at** - the date when a schedule was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**start_at** - the date when a backup started in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**duration** - how often a backup is taken

**failure_count** - the number of requests processed until the task fails

**id** - schedule ID

**period** - time period for a backup schedule (days, weeks, months, or years)

**rotation_period** - number of backups after which the first backup will be deleted

**target_id** - the disk ID for which a backup is taken

**target_type** - currently, you can schedule backup of Disks only

**user_id** - the ID of a user who created this schedule

**failure_count** - the number of requests processed until the task fails

**status** - the status of the backup schedule (enabled, disabled, or failed)

**schedule_logs** – schedule’s log messages with the following details:

- **created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **updated_at** - the date when in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **id** – log message ID
- **log_output** – log message text
- **schedule_id** – ID of a schedule to which the log message belongs to
• **status** – status of a scheduled backup performed

**Page History**

v.3.2

• added *rotation_period* parameter

### 14.3.3 Get List of Schedules for a Disk

To get a list of schedules for a particular disk, use the following request:

GET /settings/disks/:disk_id/schedules.xml
GET /settings/disks/:disk_id/schedules.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**
Where:

- **duration** - the number specifying how often a backup should be taken
- **target_id** – ID of the action target
- **period** - the time period (days, weeks, months, or years)
- **action** – the action performed
- **start_at** – time, when the action starts
- **id** – schedule id
- **user_id** – ID of the disk (action target) user
- **schedule_logs** – an array with schedule log details, where:
  - **schedule_id** – ID of a schedule
  - **id** – ID of the schedule log
  - **log_output** – an array with log details
  - **status** – status of the action (complete, failed, etc.)
  - **failure_count** – number of failures during the action
  - **status** – schedule status (enabled or disabled)
  - **target_type** – type of the target

### 14.3.4 Get List of Virtual Server Schedules

To get a list of schedules for a particular virtual server, use the following request:
GET /virtual_machines/:id/schedules.xml
GET /virtual_machines/:id/schedules.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**
<schedule type="array">
  <schedule>
    <action>autobackup</action>
    <created_at type="datetime">2014-01-20T14:23:44+02:00</created_at>
    <duration type="integer">1</duration>
    <failure_count type="integer">0</failure_count>
    <id type="integer">17</id>
    <params nil="true"/>
    <period>days</period>
    <rotation_period type="integer">1</rotation_period>
    <start_at type="datetime">2014-01-22T14:23:44+02:00</start_at>
    <status>enabled</status>
    <target_id type="integer">9</target_id>
    <target_type>VirtualMachine</target_type>
    <updated_at type="datetime">2014-01-21T14:23:49+02:00</updated_at>
    <user_id type="integer">2</user_id>
    <schedule_logs type="array">
      <schedule_log>
        <created_at type="datetime">2014-01-21T14:23:49+02:00</created_at>
        <id type="integer">39</id>
        <log_output></log_output>
        <schedule_id type="integer">17</schedule_id>
        <status>complete</status>
        <updated_at type="datetime">2014-01-21T14:23:49+02:00</updated_at>
      </schedule_log>
      <schedule_log>
        <created_at type="datetime">2014-01-20T14:23:49+02:00</created_at>
        <id type="integer">32</id>
        <log_output></log_output>
        <schedule_id type="integer">17</schedule_id>
        <status>complete</status>
        <updated_at type="datetime">2014-01-20T14:23:49+02:00</updated_at>
      </schedule_log>
    </schedule_logs>
  </schedule>
  <schedule>
    <action>autobackup</action>
    <created_at type="datetime">2014-01-20T14:23:44+02:00</created_at>
    <duration type="integer">1</duration>
    <failure_count type="integer">0</failure_count>
    <id type="integer">18</id>
    <params nil="true"/>
    <period>weeks</period>
    <rotation_period type="integer">1</rotation_period>
    <start_at type="datetime">2014-01-27T14:23:44+02:00</start_at>
    <status>enabled</status>
    <target_id type="integer">9</target_id>
    <target_type>VirtualMachine</target_type>
    <updated_at type="datetime">2014-01-20T14:23:54+02:00</updated_at>
    <user_id type="integer">2</user_id>
    <schedule_logs type="array">
      <schedule_log>
        <created_at type="datetime">2014-01-20T14:23:54+02:00</created_at>
        <id type="integer">33</id>
        <log_output></log_output>
        <schedule_id type="integer">18</schedule_id>
        <status>complete</status>
        <updated_at type="datetime">2014-01-20T14:23:54+02:00</updated_at>
      </schedule_log>
      <schedule_log>
        <created_at type="datetime">2014-01-20T14:23:54+02:00</created_at>
        <id type="integer">33</id>
        <log_output></log_output>
        <schedule_id type="integer">18</schedule_id>
        <status>complete</status>
        <updated_at type="datetime">2014-01-20T14:23:54+02:00</updated_at>
      </schedule_log>
    </schedule_logs>
  </schedule>
</schedule>
<id type="integer">19</id>
<params nil="true"/>
<period>months</period>
<rotation_period type="integer">1</rotation_period>
<start_at type="datetime">2014-02-20T14:23:44+02:00</start_at>
<status>enabled</status>
<target_id type="integer">9</target_id>
<target_type>VirtualMachine</target_type>
<updated_at type="datetime">2014-01-20T14:23:59+02:00</updated_at>
<br user_id type="integer">2</user_id>
<schedule_logs type="array">
  <schedule_log>
    <created_at type="datetime">2014-01-20T14:23:59+02:00</created_at>
    <id type="integer">34</id>
    <log_output></log_output>
    <schedule_id type="integer">19</schedule_id>
    <status>complete</status>
    <updated_at type="datetime">2014-01-20T14:23:59+02:00</updated_at>
  </schedule_log>
</schedule_logs>
</schedule>

<schedule>
  <action>autobackup</action>
  <created_at type="datetime">2014-01-20T14:23:44+02:00</created_at>
  <duration type="integer">1</duration>
  <failure_count type="integer">0</failure_count>
  <id type="integer">20</id>
  <params nil="true"/>
  <period>years</period>
  <start_at type="datetime">2015-01-20T14:23:44+02:00</start_at>
  <status>enabled</status>
  <target_id type="integer">9</target_id>
  <target_type>VirtualMachine</target_type>
  <updated_at type="datetime">2014-01-20T14:24:04+02:00</updated_at>
  <user_id type="integer">2</user_id>
  <schedule_logs type="array">
    <schedule_log>
      <created_at type="datetime">2014-01-20T14:24:04+02:00</created_at>
      <id type="integer">35</id>
      <log_output></log_output>
      <schedule_id type="integer">20</schedule_id>
      <status>complete</status>
      <updated_at type="datetime">2014-01-20T14:24:04+02:00</updated_at>
    </schedule_log>
  </schedule_logs>
</schedule>
</schedules>

Where:

duration - the number specifying how often a backup should be taken
target_id – ID of the action target
period - the time period (days, weeks, months, or years)
action – the action performed
start_at – time, when the action starts
id – schedule id
user_id – ID of the disk (action target) user
schedule_logs – an array with schedule log details, where:
- `schedule_id` – ID of a schedule
- `id` – ID of the schedule log
- `log_output` – an array with log details
- `status` – status of the action (complete, failed, etc.)
  
  *failure_count* – number of failures during the action
  
  *status* – schedule status (enabled or disabled)
  
  *target_type* – type of the target

### 14.3.5 Add Schedule to Disk

To add a schedule to a disk, use the following request:

**POST /settings/disks/:disk_id/schedules.xml**

**POST /settings/disks/:disk_id/schedules.json**

**XML Request Example**

```
```

**JSON Request Example**

```
curl -i -X POST http://onapp.test/settings/disks/3/schedules.json -d '{"schedule":{"action":"autobackup","duration":"1","period":"days","rotation_period":"1","enabled":"1","start_at": "2014-11-12 10:36"}}'  -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **action** - set autobackup to add a backup schedule
- **duration** - specify duration
- **period** - set the period (days/weeks/months/years)
- **rotation_period** - the number of backups after which the first backup will be deleted
- **enabled** - set 1 to activate a schedule, otherwise, set 0
- **start_at** - set the time when backup scheduling transaction will be created.

**Page History**

**v.3.5**

- added `start_at` parameter

**v.3.2**

- added `rotation_period` parameter

### 14.3.6 Add Schedule to Virtual Server

To add incremental backup schedule to a virtual server, use the following request:
POST /virtual_machines/:virtual_machine_id/schedules.xml
POST /virtual_machines/:virtual_machine_id/schedules.json

XML Request Example:
```
```

JSON Request Example:
```
```

Where:

**duration** - how often a disk backup is taken  
**period** - time period for a backup schedule (days, weeks, months, or years)  
**rotation_period** - number of backups after which the first backup will be deleted  
**enabled** - set 0 to deactivate a schedule. This parameter is activated (set to 1) by default.  
**action** - specify an action for the schedule (e.g. - autobackup)  
**start_at** - set the time when backup scheduling transaction will be created.

**Page History**

v.3.5
- added start_at parameter  

v.3.2
- added rotation_period parameter

**14.3.7 Edit Disk Schedule**

To edit disk's schedule, use the following request:

PUT /settings/schedules/:id.xml
PUT /settings/schedules/:id.json

XML Request Example
```
```
JSON Request Example

```
```

Where:

- **duration** - how often a disk backup is taken
- **period** - time period for a backup schedule (days, weeks, months, or years)
- **rotation_period** - number of backups after which the first backup will be deleted
- **enabled** - set 1 to activate a schedule, otherwise, set 0
- **start_at** - set the time when backup scheduling transaction will be created.

Page History

v.3.5
- added start_at parameter

v.3.2
- added rotation_period parameter

14.3.8 Edit Virtual Server Schedule

To edit virtual server's incremental backup schedule, use the following request:

PUT /virtual_machines/:id/schedules/:schedule_id.xml
PUT /virtual_machines/:id/schedules/:schedule_id.json

XML Request Example

```
```

JSON Request Example

```
```

Where:

- **duration** - how often a disk backup is taken
- **period** - time period for a backup schedule (days, weeks, months, or years)
**rotation_period** - number of backups after which the first backup will be deleted
**enabled** - set 1 to activate a schedule, otherwise, set 0
**start_at** - set the time when backup scheduling transaction will be created.

**Page History**
- v.3.5
  - added **start_at** parameter
- v.3.2
  - added **rotation_period** parameter

### 14.3.9 Delete Disk Schedule
To delete disk schedule, use the following request:

DELETE /settings/schedules/:id.xml
DELETE /settings/schedules/:id.json

**XML Request Example**
```
```

**JSON Request Example**
```
```

### 14.3.10 Delete Virtual Server Schedule
To delete a virtual server schedule, use the following request:

DELETE /settings/schedules/:id.xml
DELETE /settings/schedules/:id.json

**XML Request Example**
```
```

**JSON Request Example**
```
```

Where you have to specify backup server ID and schedule ID.
15 Backups/ Snapshots

Lists the backups/snapshots taken of that virtual server, and provides tools to restore a backup, delete backups, and convert backups to templates.

OnApp supports two backup types: normal and incremental:

- Normal backup - simple method of taking backups by making full copy of target data.
- Incremental - advanced method of taking backups. During the incremental backup, only the changes made after the last backup are archived instead of backing up the whole target.

If you are using incremental backups option, you should either enable dedicated backup servers in your cloud or share the backups and templates folders (paths) between your compute resources. SSH file transfer option will be skipped for virtual servers using incremental backups. Existing full backups will be still accessible via Backups > Images menu.

15.1 Get List of All VS Backups

To get the list of all VS backups, use the following request:

GET /virtual_machines/:virtual_machine_id/backups.xml
GET /virtual_machines/:virtual_machine_id/backups.json

XML Request Example

```
```

JSON Request Example

```
```

An array of backups is returned. If there are no backups, an empty array is returned.

XML Output Example
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <backup_size type="integer">30896</backup_size>
    <built type="boolean">true</built>
    <built_at type="datetime">2013-12-24T14:34:06+03:00</built_at>
    <created_at type="datetime">2013-12-24T14:31:20+03:00</created_at>
    <data_store_type>lvm</data_store_type>
    <id type="integer">1951</id>
    <identifier>uml64gqyvblzkv1kb</identifier>
    <image_type nil="true"/>
    <initiated>days</initiated>
    <locked type="boolean">false</locked>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <note nil="true"/>
    <operating_system>linux</operating_system>
    <operating_system_distro>ubuntu</operating_system_distro>
    <target_id type="integer">11860</target_id>
    <target_type>Disk</target_type>
    <template_id type="integer">897</template_id>
    <updated_at type="datetime">2013-12-24T14:34:06+03:00</updated_at>
    <user_id type="integer">1875</user_id>
    <volume_id nil="true"/>
    <backup_type>normal</backup_type>
    <disk_id type="integer">11860</disk_id>
  </backup>
</backups>

Where:

allowed_resize_without_reboot – true if resizing CPU & RAM is allowed without restarting the storage server backed up

allowed_hot_migrate – true if hot migration is allowed for the storage server backed up

allowed_swap – true if swap disk is allowed for storage server backed up, otherwise false

backup_server_id – the ID of the backup server on which the backup is stored

backup_size – the disk space taken by this backup in kB

backup_type – normal or incremental

built – true if the storage server backed up has been built

built_at – the date when the disk backup was built

created_at – the date when the record in the database was created

updated_at – the date when this record in database was updated

data_store_type - data store type: lvm, vmware or solidfire

id – the ID of this backup

identifier - disk identifier

image_type - backup type (currently only tar is available)
initiated - period when backup is initiated: days, weeks, months, or years
locked – true if the storage server backed up has been locked
marked_for_delete – the backup is marked for deletion (for auto-backups)
min_disk_size – the minimum disk size
operating_system_distro – the OS distribution of the storage server backed up
operating_system – the OS of the storage server backed up
target_id - ID of a backup target
target_type - target for which the backup was taken; For normal backups it is a disk. For incremental backups it’s virtual server.
template_id – the ID of a template from which the storage server backed up was built
user_id - the ID of a user the storage server belongs to
volume_id - data store ID

SolidFire - related parameters:

15.2 Get List of Normal Backups

To get the list of normal backups, use the following request:

GET /virtual_machines/:virtual_machine_id/backups/images.xml
GET /virtual_machines/:virtual_machine_id/backups/images.json

XML Request Example


JSON Request Example


An array of backups is returned. If there are no backups, an empty array is returned.

XML Output Example
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <backup_size type="integer">310896</backup_size>
    <built type="boolean">true</built>
    <built_at type="datetime">2013-12-24T14:34:06+03:00</built_at>
    <created_at type="datetime">2013-12-24T14:31:20+03:00</created_at>
    <data_store_type type="lvm">lvm</data_store_type>
    <id type="integer">1951</id>
    <identifier>uml64qygvbzv1kb</identifier>
    <image_type nil="true"/>
    <initiated>days</initiated>
    <locked type="boolean">false</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <note nil="true"/>
    <operating_system type="linux">linux</operating_system>
    <operating_system_distro type="ubuntu">ubuntu</operating_system_distro>
    <target_id type="integer">11860</target_id>
    <target_type type="Disk">Disk</target_type>
    <template_id type="integer">897</template_id>
    <updated_at type="datetime">2013-12-24T14:34:06+03:00</updated_at>
    <user_id type="integer">1875</user_id>
    <volume_id nil="true"/>
    <backup_type type="normal">normal</backup_type>
    <disk_id type="integer">11860</disk_id>
  </backup>
</backups>

Where:

- **allow_resize_without_reboot** – true if resizing CPU & RAM is allowed without restarting the storage server backed up
- **allowed_hot_migrate** – true if hot migration is allowed for the storage server backed up
- **allowed_swap** – true if swap disk is allowed for storage server backed up, otherwise false

- **backup_server_id** – the ID of the backup server on which the backup is stored
- **backup_size** – the disk space taken by this backup in kB
- **backup_type** – normal or incremental
- **built** – true if the storage server backed up has been built
- **built_at** – the date when the disk backup was built
- **created_at** – the date when the record in the database was created
- **updated_at** – the date when this record in database was updated
- **data_store_type** - data store type: lvm, vmware or solidfire
- **id** – the ID of this backup
- **identifier** - disk identifier
- **image_type** - backup type (currently only tar is available)
initiated - period when backup is initiated: days, weeks, months, or years
locked – true if the storage server backed up has been locked
marked_for_delete – the backup is marked for deletion (for auto-backups)
min_disk_size – the minimum disk size
operating_system_distro – the OS distribution of the storage server backed up
operating_system – the OS of the storage server backed up
target_id - ID of a backup target
target_type - target for which the backup was taken; For normal backups it is a disk. For incremental backups it’s virtual server.
template_id – the ID of a template from which the storage server backed up was built
user_id - the ID of a user the storage server belongs to
volume_id - data store ID

SolidFire - related parameters:

15.3 Get List of Incremental Backups

To get the list of incremental backups, use the following request:

GET /virtual_machines/:virtual_machine_id/backups/files.xml
GET /virtual_machines/:virtual_machine_id/backups/files.json

XML Request Example

```
```

JSON Request Example

```
```

An array of backups is returned. If there are no backups, an empty array is returned.

XML Output Example
<backups type="array">
  <backup type="allow_resize_without_reboot">false</type>
  <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
  <allowed_swap type="boolean">true</allowed_swap>
  <backup_server_id nil="true">
    <backup_size type="integer">1121652</backup_size>
    <built type="boolean">true</built>
    <built_at type="datetime">2013-12-18T11:00:47+00:00</built_at>
    <created_at type="datetime">2013-12-18T10:59:41+00:00</created_at>
    <data_store_type>lvm</data_store_type>
    <id type="integer">698</id>
    <identifier>y5cc19dv7bsdrk</identifier>
    <initiated>manual</initiated>
    <iqn nil="true"/>
    <locked type="boolean">false</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <note>Stuarts backup</note>
    <operating_system>linux</operating_system>
    <target_id type="integer">74</target_id>
    <target_type>VirtualMachine</target_type>
    <template_id type="integer">6</template_id>
    <updated_at type="datetime">2013-12-18T11:00:47+00:00</updated_at>
    <user_id type="integer">8</user_id>
  </backup_server_id>
  <backup_type>incremental</backup_type>
</backups>

Where:

- **allowed_resize_without_reboot** – true if resizing CPU & RAM is allowed without restarting the storage server backed up
- **allowed_hot_migrate** – true if hot migration is allowed for the storage server backed up
- **allowed_swap** – true if swap disk is allowed for storage server backed up, otherwise false
- **backup_server_id** – the ID of the backup server on which the backup is stored
- **backup_size** – the disk space taken by this backup in kB
- **backup_type** – normal or incremental
- **built** – true if the storage server backed up has been built
- **built_at** – the date when the disk backup was built
- **created_at** – the date when the record in the database was created
- **updated_at** – the date when this record in database was updated
- **data_store_type** - data store type: lvm, vmware or solidfire
- **id** – the ID of this backup
- **identifier** - disk identifier
- **image_type** - backup type (currently only tar is available)
- **initiated** - period when backup is initiated: days, weeks, months, or years
locked – true if the storage server backed up has been locked
marked_for_delete – the backup is marked for deletion (for auto-backups)
min_disk_size – the minimum disk size
operating_system_distro – the OS distribution of the storage server backed up
operating_system – the OS of the storage server backed up
target_id - ID of a backup target
target_type - target for which the backup was taken; For normal backups it is a disk. For incremental backups it's virtual server.
template_id – the ID of a template from which the storage server backed up was built
user_id - the ID of a user the storage server belongs to
volume_id - data store ID
SolidFire - related parameters:
  iqns - volume ISCSI qualified name

15.4 Get List of Disk Backups

To view the list of disk backups, use the following request:

GET /virtual_machines/:vm_id/disks/:disk_id/backups.xml
GET /virtual_machines/:vm_id/disks/:disk_id/backups.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allow_swap type="boolean">true</allow_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <backup_size type="integer">310896</backup_size>
    <built type="boolean">true</built>
    <built_at type="datetime">2013-12-24T14:34:06+03:00</built_at>
    <created_at type="datetime">2013-12-24T14:31:20+03:00</created_at>
    <data_store_type>lvm</data_store_type>
    <id type="integer">1951</id>
    <identifier>uml64qyvbzv1kb</identifier>
    <image_type nil="true"/>
    <initiated>days</initiated>
    <is_nil>true</is_nil>
    <locked type="boolean">false</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <note nil="true"/>
    <operating_system>linux</operating_system>
    <operating_system_distro>ubuntu</operating_system_distro>
    <target_id type="integer">11860</target_id>
    <target_type>Disk</target_type>
    <template_id type="integer">897</template_id>
    <updated_at type="datetime">2013-12-24T14:34:06+03:00</updated_at>
    <user_id type="integer">1875</user_id>
    <volume_id nil="true"/>
    <backup_type>normal</backup_type>
    <disk_id type="integer">11860</disk_id>
  </backup>
</backups>

Where:

allowed_resize_without_reboot – true if resizing CPU & RAM is allowed without restarting the storage server backed up
allowed_hot_migrate – true if hot migration is allowed for the storage server backed up
allowed_swap – true if swap disk is allowed for storage server backed up, otherwise false
backup_server_id – the ID of the backup server on which the backup is stored
backup_size – the disk space taken by this backup in kB
backup_type – normal or incremental
built – true if the storage server backed up has been built
built_at – the date when the disk backup was built
created_at – the date when the record in the database was created
updated_at – the date when this record in database was updated
data_store_type - data store type: lvm, vmware,solidfire or
id – the ID of this backup
identifier - disk identifier
image_type - backup type (currently only tar is available)
initiated - period when backup is initiated: days, weeks, months, or years
locked – true if the storage server backed up has been locked
marked_for_delete – the backup is marked for deletion (for auto-backups)
min_disk_size – the minimum disk size
operating_system_distro – the OS distribution of the storage server backed up
operating_system – the OS of the storage server backed up
target_id - ID of a backup target
target_type - target for which the backup was taken; For normal backups it is a disk. For incremental backups it’s virtual server.
template_id – the ID of a template from which the storage server backed up was built
user_id - the ID of a user the storage server belongs to
volume_id - data store ID

SolidFire - related parameters:
ign - volume ISCSI qualified name

15.5 Create Incremental Backup

To take an incremental backup, use the following request:

POST /virtual_machines/:id/backups.xml
POST /virtual_machines/:id/backups.json

XML Request Example

```
```

JSON Request Example

```
```

Where:

note - optional backup note

15.6 Create Disk Backup

To take incremental backups for virtual servers that have incremental backups enabled, use Create Incremental Backup API call.
To create a backup of a disk, use the following request:

POST /settings/disks/:disk_id/backups.xml
POST /settings/disks/:disk_id/backups.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**Where:**

- **note** - optional note
- **force_windows_backup** - for Windows virtual servers only; enabling this option will guarantee the disk backup will be taken regardless of any file system problems

This option should be only used as a last resort. If you enable this option there is no guarantee that backup will be consistent.

### 15.7 Create Backups for All Disks

The following request applies to configuration where normal backups are enabled for cloud, and will create backups for all disks.

To create backups/snapshots of all VS disks, use the following request:

POST /virtual_machines/:vm_id/backups.xml
POST /virtual_machines/:vm_id/backups.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```
15.8 Convert Backup to Template

You can convert a backup into a custom template. A label for a template can be set with the backup[label], minimum disk size and minimum memory size parameters.

To convert a backup into a template, use the following request:

POST /backups/:backup_id/convert.xml
POST /backups/:backup_id/convert.json

**XML Request Example**

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --data '
<backup><label>template.label</label><min_disk_size>50</min_disk_size><min_memory_size>1024</min_memory_size></backup>' --url http://onapp.test/backups/321/convert.xml
```

**JSON Request Example**

```
```

Where:

- **label** - template label
- **min_disk_size** - minimum disk size required in GB
- **min_memory_size** - minimum RAM required for the template

If templates limit has been exceeded, you will get the following error message: "You have reached your template creation limit".

15.9 Delete Backup

To delete a disk backup, use the following request:

DELETE /backups/:id.xml
DELETE /backups/:id.json

**XML Request Example**

```
```
15.10 Restore Backup

To restore a disk from a backup, use the following request:

POST /backups/:backup_id/restore.xml
POST /backups/:backup_id/restore.json

XML Request Example

curl -i -X POST -u user:userpass --url http://onapp.test/backups/4326/restore.xml -H "Content-Type: text/xml" -d '<force_restore>true</force_restore>'

JSON Request Example

curl -i -X POST -u user:userpass --url http://onapp.test/backups/4326/restore.json -H 'Content-type: application/json' -d '{"force_restore": "true"}'

Where:

- **force_restore** - true to initiate a rebuild of a file system on a disk, otherwise, false

Page History

v.5.9
- added the **force_restore** parameter

15.11 Add/Edit Backup Note

To update backup with a note, use the following request:

PUT /backups/:backup_id/note.xml
PUT /backups/:backup_id/note.json

XML Request Example
JSON Request Example

curl -X PUT http://onapp.test/backups/12/note.json -d
'{"backup":{"note":"test note"}}' -u user:password -H 'Accept: application/json' -H 'Content-type: application/json'
16 Backup Resource Auto Backup Presets

This chapter includes API requests for managing backup resource auto-backup presets.

16.1 Get List of Backup Resource Auto Backup Presets

To get the list of backup resource auto-backup presets, use the following request:

GET /settings/backups/resources/:resource_id/auto_backup_presets.xml
GET /settings/backups/resources/:resource_id/auto_backup_presets.json

XML Request Example


JSON Request Example


XML Output Example

```
<auto_backup_presets type="array">
  <auto_backup_preset>
    <created_at type="dateTime">2018-04-19T17:15:10+03:00</created_at>
    <day_to_run_on nil="true"/>
    <days_to_run_on type="array"/>
    <enabled type="boolean">true</enabled>
    <frequency type="integer">1</frequency>
    <id type="integer">14</id>
    <max_recovery_points type="integer">5</max_recovery_points>
    <period>hourly</period>
    <resource_id type="integer">5</resource_id>
    <start_time type="dateTime">2000-01-01T11:48:00Z</start_time>
    <updated_at type="dateTime">2018-04-20T11:48:40+03:00</updated_at>
  </auto_backup_preset>
</auto_backup_presets>
```

Where:

- **auto_backup_presets** - the array of auto backup presets
- **auto_backup_preset** - the array of auto backup preset parameters
- **created_at** - the date when the auto backup preset was created in the [YYYY][MM][DD][TH][hh][mm][ss] format
- **day_to_run_on** - the day when the auto backup preset is run. The parameter is applicable to the monthly period. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.
days_to_run_on - the array of days when the auto backup preset is run. The parameter is applicable to the weekly period. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.

enabled - true if the auto backup preset is enabled, otherwise, false

frequency - the frequency of how often the auto backup preset is run. The parameter is applicable to the daily period. For example, set 1 to run the auto backup preset every day, 2 - every second day, 3 - every third day, etc.

id - the ID of the auto backup preset

max_recovery_points - the maximum number of recovery points created for VS

period - the period for which the auto backup preset is configured that can be the following:

- hourly
- daily
- weekly
- monthly
- yearly

resource_id - the ID of the backup resource for which the auto backup preset is configured

start_time - the start time for running the auto backup preset in the [YYY][M][D][T][hh][mm][ss] format. The parameter is applicable to the daily, weekly and monthly periods.

updated_at - the date when the auto backup preset was updated in the [YYYY][M][D][T][hh][mm][ss] format

week_to_run_on - the week when the auto backup preset is run. The parameter is applicable to the monthly period. For example, set 0 to run the auto backup preset on the first week of the month, 1 - the second week, 2 - the third week, or 3 - the fourth week.

16.2 Get Backup Resource Auto Backup Preset Details

To get the auto backup preset details, use the following request:

GET
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.xml

GET
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```
### XML Output Example

```xml
<auto_backup_presets type="array">
    <auto_backup_preset>
        <created_at type="dateTime">2018-04-19T17:15:10+03:00</created_at>
        <day_to_run_on nil="true"/>
        <days_to_run_on type="array"/>
        <enabled type="boolean">true</enabled>
        <frequency type="integer">1</frequency>
        <id type="integer">14</id>
        <max_recovery_points type="integer">5</max_recovery_points>
        <period>hourly</period>
        <resource_id type="integer">5</resource_id>
        <start_time type="dateTime">2000-01-01T11:48:00Z</start_time>
        <updated_at type="dateTime">2018-04-20T11:48:40+03:00</updated_at>
    </auto_backup_preset>
</auto_backup_presets>
```

### Where:

- **auto_backup_presets** - the array of auto backup presets
- **auto_backup_preset** - the array of auto backup preset parameters
- **created_at** - the date when the auto backup preset was created in the `[YYYY][MM][DD][T][hh][mm][ss]` format
- **day_to_run_on** - the day when the auto backup preset is run. The parameter is applicable to the **monthly** period. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.
- **days_to_run_on** - the array of days when the auto backup preset is run. The parameter is applicable to the **weekly** period. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.
- **enabled** - **true** if the auto backup preset is enabled, otherwise, **false**
- **frequency** - the frequency of how often the auto backup preset is run. The parameter is applicable to the **daily** period. For example, set 1 to run the auto backup preset every day, 2 - every second day, 3 - every third day, etc.
- **id** - the ID of the auto backup preset
- **max_recovery_points** - the maximum number of recovery points created for VS
- **period** - the period for which the auto backup preset is configured that can be the following:
  - **hourly**
  - **daily**
  - **weekly**
  - **monthly**
  - **yearly**
- **resource_id** - the ID of the backup resource for which the auto backup preset is configured
- **start_time** - the start time for running the auto backup preset in the `[YYYY][MM][DD][T][hh][mm][ss]` format
- **updated_at** - the date when the auto backup preset was updated in the `[YYYY][MM][DD][T][hh][mm][ss]` format
- **week_to_run_on** - the week when the auto backup preset is run. The parameter is applicable to the **monthly** period. For example, set 0 to run the auto backup preset on the first week of the month, 1 - the second week, 2 - the third week, or 3 - the fourth week.
16.3 Add Backup Resource Hourly Auto Backup Preset

To add an hourly auto backup preset, use the following request:

POST /settings/backups/resources/:resource_id/auto_backup_presets.xml
POST /settings/backups/resources/:resource_id/auto_backup_presets.json

XML Request Example

```curl
<auto_backup_preset>
  <resource_id type="integer">3</resource_id>
  <period>hourly</period>
  <max_recovery_points type="integer">1</max_recovery_points>
</auto_backup_preset>
'```

JSON Request Example

```curl
  "auto_backup_preset": {
    "resource_id": 9,
    "period": "hourly",
    "max_recovery_points": 1
  }
}'```

Where:
- **auto_backup_preset** - the array of auto backup preset parameters
- **resource_id** - the ID of the backup resource for which the auto backup preset is configured
- **period** - the period for the auto backup preset that is hourly
- **max_recovery_points** - the maximum number of recovery points created for VS

16.4 Add Backup Resource Daily Auto Backup Preset

To add a daily auto backup preset, use the following request:

POST /settings/backups/resources/:resource_id/auto_backup_presets.xml
POST /settings/backups/resources/:resource_id/auto_backup_presets.json

XML Request Example

```curl
<auto_backup_preset>
  <resource_id type="integer">13</resource_id>
  <period>daily</period>
  <frequency type="integer">2</frequency>
  <max_recovery_points type="integer">1</max_recovery_points>
  <start_time>2000-01-01T11:48:00Z</start_time>
</auto_backup_preset>
'```

JSON Request Example

```curl
  "auto_backup_preset": {
    "resource_id": 13,
    "period": "daily",
    "frequency": 2,
    "max_recovery_points": 1,
    "start_time": 2000-01-01T11:48:00Z
  }
}'```

Where:

auto_backup_preset - the array of auto backup preset parameters
resource_id* - the ID of the backup resource for which the auto backup preset is configured
period* - the period for the auto backup preset that is daily
frequency* - the frequency of how often the auto backup preset is run. For example, set 1 to run the auto backup preset every day, 2 - every second day, 3 - every third day, etc.
max_recovery_points - the maximum number of recovery points created for VS
start_time - the start time for running the auto backup preset in the [YYYY][MM][DD][hh][mm][ss] format

16.5 Add Backup Resource Weekly Auto Backup Preset

To add a weekly auto backup preset, use the following request:
POST /settings/backups/resources/:resource_id/auto_backup_presets.xml
POST /settings/backups/resources/:resource_id/auto_backup_presets.json

XML Request Example

curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/backups/resources/13/auto_backup_presets.xml -d '<auto_backup_preset><resource_id type="integer">13</resource_id><period>weekly</period><days_to_run_on><fixnum type="integer">1</fixnum><fixnum type="integer">2</fixnum></days_to_run_on><max_recovery_points type="integer">1</max_recovery_points><start_time>2000-01-01T11:48:00Z</start_time></auto_backup_preset>'

JSON Request Example

curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/backups/resources/13/auto_backup_presets.json -d '{"auto_backup_preset":{"resource_id": 11,"period": "weekly","days_to_run_on": [1,2,3],"max_recovery_points": 1,"start_time": "2000-01-01T11:48:00Z"}}'

Where:

auto_backup_preset - the array of auto backup preset parameters
resource_id* - the ID of the backup resource for which the auto backup preset is configured
period* - the period for the auto backup preset that is weekly
days_to_run_on* - the array of days when the auto backup preset is run. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.

max_recovery_points - the maximum number of recovery points created for VS

start_time - the start time for running the auto backup preset in the [YYYY][MM][DD][hh][mm][ss] format

16.6 Add Backup Resource Monthly Auto Backup Preset

To add a monthly auto backup preset, use the following request:

POST /settings/backups/resources/:resource_id/auto_backup_presets.xml
POST /settings/backups/resources/:resource_id/auto_backup_presets.json

XML Request Example

curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/backups/resources/11/auto_backup_presets.xml -d '<auto_backup_preset><resource_id type="integer">11</resource_id><period>monthly</period><day_to_run_on type="integer">1</day_to_run_on><week_to_run_on type="integer">1</week_to_run_on><max_recovery_points type="integer">1</max_recovery_points><start_time>2000-01-01T11:48:00Z</start_time></auto_backup_preset>'

JSON Request Example


Where:

auto_backup_preset - the array of auto backup preset parameters
resource_id* - the ID of the backup resource for which the auto backup preset is configured
period* - the period for the auto backup preset that is monthly
day_to_run_on* - the day when the auto backup preset is run. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.
week_to_run_on* - the week when the auto backup preset is run. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.
max_recovery_points - the maximum number of recovery points created for VS
start_time - the start time for running the auto backup preset in the [YYYY][MM][DD][hh][mm][ss] format
16.7 Add Backup Resource Yearly Auto Backup Preset

To add a yearly auto backup preset, use the following request:

POST /settings/backups/resources/:resource_id/auto_backup_presets.xml
POST /settings/backups/resources/:resource_id/auto_backup_presets.json

**XML Request Example**

```
  <!--auto_backup_preset-->
  <auto_backup_preset><resource_id type="integer">12</resource_id><period>yearly</period><max_recovery_points type="integer">1</max_recovery_points><day_to_run_on type="integer">1</day_to_run_on><start_time>2000-01-01T11:48:00Z</start_time>-->

```  

**JSON Request Example**

```
  "auto_backup_preset": {
    "resource_id": 12,
    "period": "yearly",
    "day_to_run_on": 1,
    "start_time": "2000-01-01T11:48:00Z",
    "max_recovery_points": 1
  }
}'
```

Where:

- `auto_backup_preset` - the array of auto backup preset parameters
- `resource_id` - the ID of the backup resource for which the auto backup preset is configured
- `period` - the period for the auto backup preset that is yearly
- `max_recovery_points` - the maximum number of recovery points created for VS
- `day_to_run_on` - the day when the auto backup preset is run. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.
- `start_time` - the start time for running the auto backup preset in the [YYYY][MM][DD][HH][mm][ss] format

**Page History**

v.6.1 Edge 2

- added the following parameters:
  - `day_to_run_on`
  - `start_time`

16.8 Edit Backup Resource Hourly Auto Backup Preset

To edit an hourly auto backup preset, use the following request:
PUT
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.xml

PUT
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

Where:

- `auto_backup_preset` - the array of auto backup preset parameters
- `resource_id` - the ID of the backup resource for which the auto backup preset is configured
- `max_recovery_points` - the maximum number of recovery points created for VS

16.9 Edit Backup Resource Daily Auto Backup Preset

To edit a daily auto backup preset, use the following request:

PUT
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.xml

PUT
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.json

XML Request Example

```bash
```

JSON Request Example
16.10 Edit Backup Resource Weekly Auto Backup Preset

To edit a weekly auto backup preset, use the following request:

```
PUT /settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.xml
PUT /settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.json
```

**XML Request Example**

```
-d '<auto_backup_preset><resource_id><fixnum type="integer">13</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">2</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnum type="integer">1</fixnum><fixnu
```
days_to_run_on* - the array of days when the auto backup preset is run. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.

max_recovery_points* - the maximum number of recovery points created for VS

start_time - the start time for running the auto backup preset in the [YYYYMMDD][HH][MM][SS] format

16.11 Edit Backup Resource Monthly Auto Backup Preset

To edit a monthly auto backup preset, use the following request:

PUT
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.xml

PUT /settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.json

XML Request Example

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/settings/backups/resources/11/auto_backup_presets/45.xml
-d 'auto_backup_preset><resource_id type="integer">11</resource_id><day_to_run_on type="integer">1</day_to_run_on><week_to_run_on type="integer">1</week_to_run_on><max_recovery_points type="integer">1</max_recovery_points><start_time>2000-01-01T11:48:00Z</start_time></auto_backup_preset>'

JSON Request Example

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/settings/backups/resources/11/auto_backup_presets/45.json
-d '{"auto_backup_preset":{"resource_id": 11,"day_to_run_on": 1,"week_to_run_on": 1,"max_recovery_points": 1,"start_time": "2000-01-01T11:48:00Z"}}'

Where:

auto_backup_preset* - the array of auto backup preset parameters

resource_id* - the ID of the backup resource for which the auto backup preset is configured
day_to_run_on* - the day when the auto backup preset is run. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.
week_to_run_on* - the week when the auto backup preset is run. For example, set 0 to run the auto backup preset on the first week of the month, 1 - the second week, 2 - the third week, or 3 - the fourth week.
max_recovery_points* - the maximum number of recovery points created for VS

start_time* - the start time for running the auto backup preset in the [YYYYMMDD][HH][MM][SS] format
16.12 Edit Backup Resource Yearly Auto Backup Preset

To edit a yearly auto backup preset, use the following request:

PUT
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.xml

PUT
/settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.json

XML Request Example

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/settings/backups/resources/12/auto_backup_presets/45.xml
-d '<auto_backup_preset><resource_id type="integer">12</resource_id><max_recovery_points type="integer">1</max_recovery_points><day_to_run_on type="integer">1</day_to_run_on><start_time>2000-01-01T11:48:00Z</start_time>'

JSON Request Example

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/settings/backups/resources/12/auto_backup_presets/45.json
-d '{"auto_backup_preset":{"resource_id": 12,"day_to_run_on": 1,"start_time": "2000-01-01T11:48:00Z","max_recovery_points": 1}}'

Where:

auto_backup_preset - the array of auto backup preset parameters
resource_id - the ID of the backup resource for which the auto backup preset is configured
max_recovery_points - the maximum number of recovery points created for VS

💡 day_to_run_on - the day when the auto backup preset is run. For example, set 0 to run the auto backup preset on Sunday, 1 - Monday, 2 - Tuesday, 3 - Wednesday, 4 - Thursday, 5 - Friday, or 6 - Saturday.

💡 start_time - the start time for running the auto backup preset in the [YYYY][MM][DD][hh][mm][ss] format

Page History

v. 6.1 Edge 2

- added the following parameters:
  - day_to_run_on
  - start_time

16.13 Delete Backup Resource Auto Backup Preset

To delete an auto backup preset, use the following request:
DELETE /settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.xml

DELETE /settings/backups/resources/:resource_id/auto_backup_presets/:auto_backup_preset_id.json

XML Request Example


JSON Request Example

17 Backup Resources

This chapter includes API request for creating and managing backup resources.

17.1 Get List of Backup Resources

To get the list of backup resources, use the following request:

GET /settings/backups/resources.xml
GET /settings/backups/resources.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<resources type="array">
  <resource>
    <advanced_options>
      <option_1>9</option_1>
      <option_2>8</option_2>
    </advanced_options>
    <resource_zone_id type="integer">3</resource_zone_id>
    <created_at type="dateTime">2018-04-04T12:01:31+03:00</created_at>
    <enabled type="boolean">true</enabled>
    <id type="integer">4</id>
    <label>backup_resource</label>
    <password>password</password>
    <plugin>r1soft</plugin>
    <primary_host>http://69.169.247.107:9080</primary_host>
    <secondary_host>http://69.168.257.106:9080</secondary_host>
    <updated_at type="dateTime">2018-04-04T12:17:27+03:00</updated_at>
    <username>username</username>
  </resource>
  ...
</resources>
```

Where:

- **resource** - the array of parameters for the backup resource
- **advanced_options** - the array of advanced options for the backup resource
- **resource_zone_id** - the ID of the backup resource zone to which the backup resource is assigned
created_at - the date when the backup resource was created in the [YYYY][MM][DD][hh][mm][ss] format

enabled - the status that indicates whether the backup resource is enabled (true) or not (false)

id - the ID of the backup resource

label - the label of the backup resource

password - the password used to connect to the third-party backup system

plugin - the label of the backup plugin

primary_host - the primary address (either hostname or IP address) used to connect to the third-party backup system

secondary_host - the secondary address (either hostname or IP address) used to connect to the third-party backup system

updated_at - the date when the backup resource was updated in the [YYYY][MM][DD][hh][mm][ss] format

username - the username used to connect to the third-party backup system

### 17.2 Get Backup Resource Details

To get the backup resource details, use the following request:

GET /settings/backups/resources/:resource_id.xml
GET /settings/backups/resources/:resource_id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
<resources type="array">
  <resource>
    <advanced_options>
      <option_1>9</option_1>
      <option_2>8</option_2>
    </advanced_options>
    <resource_zone_id type="integer">3</resource_zone_id>
    <created_at type="dateTime">2018-04-04T12:01:31+03:00</created_at>
    <enabled type="boolean">true</enabled>
    <id type="integer">4</id>
    <label>backup_resource</label>
    <password>password</password>
    <plugin>r1soft</plugin>
    <primary_host>http://69.169.247.107:9080</primary_host>
    <secondary_host>http://69.168.257.106:9080</secondary_host>
    <updated_at type="dateTime">2018-04-04T12:17:27+03:00</updated_at>
    <username>username</username>
  </resource>
</resources>

Where:

resource - the array of parameters for the backup resource

advanced_options - the array of advanced options for the backup resource

resource_zone_id - the ID of the backup resource zone to which the backup resource is assigned

created_at - the date when the backup resource was created in the [YYYY][MM][DD][hh][mm][ss] format

enabled - the status that indicates whether the backup resource is enabled (true) or not (false)

id - the ID of the backup resource

label - the label of the backup resource

password - the password used to connect to the third-party backup system

plugin - the label of the backup plugin

primary_host - the primary address (either hostname or IP address) used to connect to the third-party backup system

secondary_host - the secondary address (either hostname or IP address) used to connect to the third-party backup system

updated_at - the date when the backup resource was updated in the [YYYY][MM][DD][hh][mm][ss] format

username - the username used to connect to the third-party backup system

17.3 Add Backup Resource

To add a backup resource, use the following request:

POST /settings/backups/resources.xml
POST /settings/backups/resources.json

XML Request Example
**curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/backups/resources.xml -d '<<resource><label>resource_label</label><enabled>true</enabled><plugin>backup_plugin_label</plugin><primary_host>1.1.1.1</primary_host><secondary_host>2.2.2.2</secondary_host><username>username</username><password>password</password><resource_zone_id type="integer">1</resource_zone_id></resource>''

**JSON Request Example**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/backups/resources.json -d '{"resource":{"label":"resource_label","enabled":"true","plugin":"plugin_label","primary_host":"1.1.1.1","secondary_host":"2.2.2.2","username":"username","password":"password","resource_zone_id":1}}'```

**Where:**

- `resource` - the array of parameters for the backup resource
- `label`* - the label of the backup resource
- `enabled` - set `true` to enable the backup resource, otherwise, `false`
- `plugin`* - the label of the backup plugin
- `primary_host`* - the primary address (either hostname or IP address) used to connect to the third-party backup system
- `secondary_host` - the secondary address (either hostname or IP address) used to connect to the third-party backup system
- `username`* - the username used to connect to the third-party backup system
- `password`* - the password used to connect to the third-party backup system
- `resource_zone_id`* - the ID of the backup resource zone to assign the backup resource to

**17.4 Edit Backup Resource**

To edit a backup resource, use the following request:

PUT /settings/backups/resources/:resource_id.xml

PUT /settings/backups/resources/:resource_id.json

**XML Request Example**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/backups/resources/8.xml -d '<<resource><label>backup_resource_label</label><enabled>true</enabled><primary_host>1.1.1.1</primary_host><secondary_host>2.2.2.2</secondary_host><username>username</username><password>password</password></resource>''
```
### JSON Request Example

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/backups/resources/8.json -d '{"resource":{"label":"resource_label","enabled":"true","primary_host":"1.1.1.1","secondary_host":"2.2.2.2","username":"username","password":"password"}}'
```

**Where:**
- `resource` - the array of parameters for the backup resource
  - `label` - the label of the backup resource
  - `enabled` - set `true` to enable the backup resource, otherwise, `false`
  - `primary_host` - the primary address (either hostname or IP address) used to connect to the third-party backup system
  - `secondary_host` - the secondary address (either hostname or IP address) used to connect to the third-party backup system
  - `username` - the username used to connect to the third-party backup system
  - `password` - the password used to connect to the third-party backup system

### 17.5 Edit Backup Resource Advanced Options

To edit the backup resource advanced options, use the following request:

PUT /settings/backups/resources/:resource_id/advanced_options.xml

PUT /settings/backups/resources/:resource_id/advanced_options.json

**XML Request Example**

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/backups/resources/8/advanced_options.xml -d '<advanced_options><option_1>9</option_1><option_2>8</option_2></advanced_options>'
```

**JSON Request Example**

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/backups/resources/8/advanced_options.json -d '{"advanced_options":{"option_1":"9","option_2":"8"}}'
```

**Where:**
- `advanced_options` - the array of advanced options for the backup resource

There are the following backup advanced options that you can edit for the Veeam plugin:
- `vsphere_template_job_name` - a name of a vSphere backup job that is added to the Veeam backup infrastructure
backup_repository_name - a name of a backup repository that is added to the Veeam backup infrastructure

d power_on_after_restore - set to true if you want to power on virtual servers after restoring them from recovery points, otherwise, false

quick_rollback - set to true if you want to perform an incremental restore of virtual servers from recovery points, otherwise, false. For more information on the incremental restore, see Quick Rollback.

Page History
v. 5.9
• added the following parameters for Veeam-based backup resources:
  o vsphere_template_job_name
  o backup_repository_name
  o power_on_after_restore
  o quick_rollback

17.6 Delete Backup Resource

To delete a backup resource, use the following request:

DELETE /settings/backups/resources/:resource_id.xml
DELETE /settings/backups/resources/:resource_id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```
18 Backup Resource Zones

This chapter includes API request for creating and managing backup resource zones.

18.1 Get List of Backup Resource Zones

To get the list of backup resource zones, use the following request:

GET /settings/backups/resource_zones.xml
GET /settings/backups/resource_zones.json

XML Request Example


JSON Request Example


XML Output Example

<resource_zones type="array">
  <resource_zone>
    <created_at type="dateTime">2018-03-27T18:12:40+03:00</created_at>
    <id type="integer">1</id>
    <label>backup_resource_zone</label>
    <location_group_id>2</location_group_id>
    <updated_at type="dateTime">2018-03-28T14:58:33+03:00</updated_at>
  </resource_zone>
</resource_zones>

Where:

resource_zone - the array of parameters for the backup resource zone
created_at - the date when the backup resource zone was created in the [YYYY][MM][DD][hh][mm][ss] format
id - the ID of the backup resource zone
label - the label of the backup resource zone
location_group_id - the ID of the location group to which the backup resource zone is added
updated_at - the date when the backup resource zone was updated in the [YYYY][MM][DD][hh][mm][ss] format
18.2 Get Backup Resource Zone Details

To get the backup resource zone details, use the following request:

GET /settings/backups/resource_zones/:resource_zone_id.xml
GET /settings/backups/resource_zones/:resource_zone_id.json

XML Request Example


JSON Request Example


XML Output Example

<resource_zones type="array">
    <resource_zone>
        <created_at type="dateTime">2018-03-27T18:12:40+03:00</created_at>
        <id type="integer">1</id>
        <label>backup_resource_zone</label>
        <location_group_id>2</location_group_id>
        <updated_at type="dateTime">2018-03-28T14:58:33+03:00</updated_at>
    </resource_zone>
</resource_zones>

Where:

resource_zone - the array of parameters for the backup resource zone

created_at - the date when the backup resource zone was created in the [YYYY][MM][DD][T][hh][mm][ss] format
id - the ID of the backup resource zone
label - the label of the backup resource zone
location_group_id - the ID of the location group to which the backup resource zone is added
updated_at - the date when the backup resource zone was updated in the [YYYY][MM][DD][T][hh][mm][ss] format

18.3 Add Backup Resource Zone

To add a backup resource zone, use the following request:

POST /settings/backups/resource_zones.xml
POST /settings/backups/resource_zones.json

XML Request Example
OnApp Cloud 6.3 API Guide

**18.4 Edit Backup Resource Zone**

To edit a backup resource zone, use the following request:

PUT /settings/backups/resource_zones/:resource_zone_id.xml
PUT /settings/backups/resource_zones/:resource_zone_id.json

**XML Request Example**

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/backups/resource_zones/1.xml -d '<resource_zone><label>resource_zone_label</label><location_group_id>3<location_group_id></resource_zone>'
```

**JSON Request Example**

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/backups/resource_zones/1.json -d '{"resource_zone":{"label": "resource_zone_label","location_group_id": 3}}'
```

Where:

- **resource_zone** - the array of parameters for the backup resource zone
- **label** - the label of the backup resource zone
- **location_group_id** - the ID of the location group where the backup resource zone will reside
18.5 Add Backup Resource to Backup Resource Zone

To add a backup resource to a backup resource zone, use the following request:

```plaintext
POST /settings/backups/resource_zones/:resource_zone_id/resources/:resource_id/attach.xml
```

```plaintext
POST /settings/backups/resource_zones/:resource_zone_id/resources/:resource_id/attach.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

18.6 Remove Backup Resource from Backup Resource Zone

To remove a backup resource from a backup resource zone, use the following request:

```plaintext
POST /settings/backups/resource_zones/:resource_zone_id/resources/:attached_resource_id/detach.xml
```

```plaintext
POST /settings/backups/resource_zones/:resource_zone_id/resources/:attached_resource_id/detach.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```
18.7 Delete Backup Resource Zone

To delete a backup resource zone, use the following request:

DELETE /settings/backups/resource_zones/:resource_zone_id.xml
DELETE /settings/backups/resources_zones/:resource_zone_id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```
19 Backup Servers

The backup servers feature allows users to store their backups and templates on the backup servers set up in the cloud. Backup servers can be organized into backup server zones. All API calls are available to this class.

19.1 Get List of Backup Servers

To get the list of all backup servers in the cloud, use the following request:

GET /settings/backup_servers.xml
GET /settings/backup_servers.json

**XML Request Example**

```
```

**JSON Request Example**

```
```
<backup_servers type="array">
  <backup_server>
    <label>bk1</label>
    <created_at type="datetime">2012-01-04T10:18:59Z</created_at>
    <updated_at type="datetime">2012-01-16T14:11:30Z</updated_at>
    <backup_server_group_id type="integer">28</backup_server_group_id>
    <id type="integer">1</id>
    <backup_ip_address>192.168.123.1</backup_ip_address>
    <enabled type="boolean">true</enabled>
  </backup_server>
  <backup type="array">
    <marked_for_delete type="boolean">false</marked_for_delete>
    <disk_id type="integer">3908</disk_id>
    <built_at type="datetime">2012-02-09T16:05:21Z</built_at>
    <operating_system_distro>rhel</operating_system_distro>
    <created_at type="datetime">2012-02-09T16:03:45Z</created_at>
    <template_id type="integer">233</template_id>
    <operating_system_linux>linux</operating_system_linux>
    <updated_at type="datetime">2012-02-09T16:05:21Z</updated_at>
    <backup_type>normal</backup_type>
    <allowed_swap type="boolean">true</allowed_swap>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <id type="integer">1508</id>
    <backup_server_id type="integer">1</backup_server_id>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <min_disk_size type="integer">5</min_disk_size>
    <identifier>pkg0v4k4n34ym8</identifier>
    <locked type="boolean">false</locked>
    <built type="boolean">true</built>
  </backup>
  </backup_server>
</backup_servers>

Where:

**Backup server parameters:**

- **label** – the backup server label
- **created_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **backup_server_group_id** – the ID of a backup server group the backup server belongs to
- **id** – the backup server ID
- **backup_server_ip_address** – provisioning network IP address
- **enabled** – backup server parameter; if "enabled" = true, the backup server is enabled; if enabled = false, the backup server is disabled.

**Backup parameters:**

- **marked_for_delete** – the backup is marked for deletion (for auto-backups)
- **disk_id** – the ID of a disk backed up
- **built_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **operating_system_distro** – the OS distribution of the VS from which the backup was created
- **created_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **template_id** – the ID of the template the VS is based on
- **updated_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
backup_type – disk backup type (normal, Days auto-backup, Weeks auto-backup, Months auto-backup, Years auto-backup)

allowed_swap – true if the template to which the backup can be restored will allow swap, otherwise false.

allow_resize_without_reboot – true if the template to which the backup can be restored will support resize without reboot option, otherwise false

id – the backup ID

backup_server_id – the ID of the backup server on which the backup is stored.

allowed_hot_migrate – true if the template to which the backup can be restored will support hot migration, otherwise false.

backup_size – the size of the backup

min_disk_size – minimum disk size required for restoring a backup

identifier – the backup identifier

locked – true if the backup is being built, otherwise false

built – true if the backup is already built, otherwise false

19.2 Get Backup Server Details

To get the details for a particular backup server, use the following request:

GET /settings/backup_servers/:id.xml
GET /settings/backup_servers/:id.json

XML Request Example


JSON Request Example


XML Output Example
<backup_server>
  <label>bkl</label>
  <created_at type="datetime">2012-01-04T12:04:10:18:59+02:00592</created_at>
  <backups type="array">
    <backup>
      <marked_for_delete type="boolean">false</marked_for_delete>
      <disk_id type="integer">4097</disk_id>
      <created_at type="datetime">2012-02-11T00:36:17Z</created_at>
      <operating_system_distro>rhel</operating_system_distro>
      <template_id type="integer">211</template_id>
      <operating_system>linux</operating_system>
      <built_at nil="true"/>
      <operating_system>rhel</operating_system>
      <created_at type="datetime">2012-03-05T13:42:15+02:0002-11T00:36:17Z</created_at>
      <backup_type>months-autobackup</backup_type>
      <allowed_swap type="boolean">true</allowed_swap>
      <allow_resize_without_reboot type="boolean">true</allow_resize_without_reboot>
      <id type="integer">1526</id>
      <backup_server_id type="integer">1</backup_server_id>
      <allowed_hot_migrate type="integer">1</allowed_hot_migrate>
      <backup_size nil="true"/>
      <min_disk_size nil="true"/>
      <identifier>gmkrf5s0s4hsnj</identifier>
      <locked type="boolean">true</locked>
      <built type="boolean">false</built>
    </backup>
  </backups>
  <updated_at type="datetime">2012-02-14T14:01:20Z</updated_at>
  <backup_ip_address>192.168.123.1</backup_ip_address>
  <backup_server_group_id nil="true" type="integer">28</backup_server_group_id>
  <id type="integer">1</id>
  <enabled type="boolean">true</enabled>
  <capacity type="integer">460</capacity>
  <ip_address>109.123.105.162</ip_address>
</backup_server>

Where:

- **label** – backup server label
- **created at** – the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **updated at** – the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **id** – the backup server ID
- **backups** – a list of backups stored on this backup server ID with the following details:
  - **template_id** – the ID of the template the VS from which the backup was created was based on
  - **operating_system** – the operating system of the VA from which the backup was created
  - **backup_server_group_id** – the ID of the backup server zone the backup server belongs to

- **enabled** – backup server parameter; if "enabled" = true, the backup server will be enabled; if enabled=false, the backup server is will be disabled.
- **capacity** – the backup server capacity
- **ip_address** – the backup server IP
- **backup_server_ip_address** - provisioning network IP address
To view the list of backups with their details, please refer to the Search Backups section.

19.3 Get Integrated Storage Settings Details

To view the details of integrated storage settings for a particular backup server, use the following request:

GET /settings/backup_servers/:backup_server_id/integrated_storage_settings.xml
GET /settings/backup_servers/:backup_server_id/integrated_storage_settings.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```
<integrated_storage_settings>
  <bonding_mode>802.3ad</bonding_mode>
  <mtu type="integer">1500</mtu>
  <vlan nil="integer">123</vlan>
</integrated_storage_settings>
```

Where:

- **bonding_mode** - the type of bonding mode
- **mtu** - the maximum transportation unit size
- **vlan** - the ID of a VLAN number

19.4 Add Backup Server

To create a backup server where users will be able to store backups and templates, use the following request:

POST /settings/backup_servers.xml
POST /settings/backup_servers.json
XML Request Example

```
curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_servers.xml -d
  "<backup_server><label>az_val</label><enabled>1</enabled><capacity>40</capacity><ip_address>172.0.0.1</ip_address><backup_ip_address>192.168.123.1</backup_ip_address><integrated_storage>1</integrated_storage></backup_server>" -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_servers.json -d
  '{"backup_server":{"label":"az_val", "enabled":"1", "capacity":"40", "ip_address":"172.0.0.1","backup_ip_address":"192.168.123.1", "integrated_storage":"1"}}' -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **label** – backup server label
- **ip_address** – valid Ipv4 address
- **backup_server_ip_address** - provisioning network IP address
- **capacity** – set the backup server capacity
- **enabled** – set the "enabled" as 1 if you want the backup server to be enabled, or 0 if you want it to be disabled. If you skip the enabled parameter, the backup server will be disabled by default.
- **integrated_storage** - set to 1 to enable integrated storage on the backup server, or 0 if you want it to be disabled. If you skip this parameter, the integrated storage will be disabled by default.

XML Output Example

```
<backup_server>
  <label>az_val_1</label>
  <created_at type="datetime">2012-02-10T15:14:53Z</created_at>
  <updated_at type="datetime">2012-02-10T15:14:53Z</updated_at>
  <backup_ip_address>192.168.123.1</backup_ip_address>
  <backup_server_group_id nil="true"></backup_server_group_id>
  <id type="integer">25</id>
  <enabled type="boolean">true</enabled>
  <capacities type="array">40</capacities>
  <ip_address>172.0.0.1</ip_address>
  <integrated_storage type="boolean">false</integrated_storage>
</backup_server>
```

Page History

- v.6.1 Edge 2
  - added the integrated_storage parameter
19.5 Edit Backup Server

To edit a backup server, use the following request:

PUT /settings/backup_server/:id.xml
PUT /settings/backup_server/:id.json

XML Request Example

curl -i -X PUT -u user:userpass http://onapp.test/settings/backup_servers/2.xml -d '
<backup_server><label>az_val_ue_xml</label><enabled>1</enabled><capacity>40</capacity><ip_address>172.0.0.2</ip_address><backup_ip_address>192.168.123.1</backup_ip_address><integrated_storage>1</integrated_storage></backup_server>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X PUT -u user:userpass http://onapp.test/settings/backup_servers/2.json -d '{"backup_server":{"label":"az_val_ue_json", "enabled":"1", "capacity":"40", "ip_address":"172.0.0.1","backup_ip_address":"192.168.123.1","integrated_storage":"1"}}' -H 'Accept: application/json' -H 'Content-type: application/json'

Where:

- **label** – backup server label
- **enabled** – set the "enabled" as 1 if you want the backup server to be enabled, or 0 if you it to be disabled. If you skip the **enabled** parameter, the backup server will be disabled by default.
- **capacity** – set the backup server capacity
- **ip_address** – valid IPv4 address
- **backup_server_ip_address** - provisioning network IP address
- **integrated_storage** - set to 1 to enable integrated storage on the backup server, or 0 if you want it to be disabled. If you skip this parameter, the integrated storage will be disabled by default.

Page History

v6.1 Edge 2
- added the **integrated_storage** parameter

19.6 Edit Integrated Storage Settings

To edit integrated storage settings, use the following request:

PUT /settings/backup_servers/:backup_server_id/integrated_storage_settings.xml
PUT
OnApp Cloud 6.3 API Guide

/settings/backup_servers/:backup_server_id/integrated_storage_settings.json

XML Request Example

curl -i -X -u user:userpass PUT
http://onapp.test/settings/backup_servers/1/integrated_storage_settings.xml
- H 'Accept: application/xml' - H 'Content-type: application/xml'

JSON Request Example

curl -i -X -u user:userpass PUT
http://onapp.test/settings/backup_servers/1/integrated_storage_settings.json
- H 'Accept: application/json' - H 'Content-type: application/json'

Where:

bonding_mode - the type of bonding mode
mtu - the maximum transportation unit size
vlan - the ID of a VLAN number

19.7 Delete Backup Server

To delete a backup server, use the following request:

DELETE /settings/backup_servers/:id.xml
DELETE /settings/backup_servers/:id.json

XML Request Example

curl -i -X DELETE -u user:userpass
http://onapp.test/settings/backup_servers/2.xml
- H 'Accept: application/xml' - H 'Content-type: application/xml'

JSON Request Example

curl -i -X DELETE -u user:userpass
http://onapp.test/settings/backup_servers/2.json
- H 'Accept: application/json' - H 'Content-type: application/json'

You will get a 204 status response on success, and 404 if there is no such a backup server with a requested ID or you entered incorrect URL.

19.8 Search Backups

To find a backup stored at a particular backup server, use the following request:

GET /settings/backup_servers/:id/backups_search.xml
GET /settings/backup_servers/:id/backups_search.json

XML Request Example

JSON Request Example


Where:
size – set the size [from] and [to] backup search parameters to search backups by their size. The size should be indicated in MB.
date – set the date [startdate] and [enddate] backup search parameters to search for backups created between two dates. The date should be indicated in the YYYY-MM-DD format.

Please be aware that some Unix command shells can output an error because of square brackets. To prevent the error, use the backslash escape symbol. The example curl with backslashes is as follows:

XML Request Example


JSON Request Example


XML Output Example
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">true</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <backup_size type="integer">4976984</backup_size>
    <built type="boolean">true</built>
    <built_at type="datetime">2016-09-21T12:26:11Z</built_at>
    <created_at type="datetime">2016-09-21T12:08:06Z</created_at>
    <data_store_type type="string">lvm</data_store_type>
    <id type="integer">872</id>
    <identifier type="string">dyhy150m</identifier>
    <initiated type="string">manual</initiated>
    <iqn nil="true"/>
    <locked type="boolean">false</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">9</min_disk_size>
    <min_memory_size type="integer">384</min_memory_size>
    <note type="string">zaza50patch76</note>
    <operating_system type="string">linux</operating_system>
    <operating_system_distro type="string">rhel</operating_system_distro>
    <target_id type="integer">9287</target_id>
    <target_type type="string">Disk</target_type>
    <template_id type="integer">28</template_id>
    <updated_at type="datetime">2016-09-21T12:26:15Z</updated_at>
    <user_id type="integer">3</user_id>
    <volume_id nil="true"/>
    <backup_type type="string">normal</backup_type>
    <disk_id type="integer">9287</disk_id>
  </backup>
  ...
</backups>

Where:

allow_resize_without_reboot - true if the template to which the backup can be restored will support resize without reboot option, otherwise false

allowed_hot_migrate - true if the template to which the backup can be restored will support hot migration, otherwise false.

allowed_swap - true if the template to which the backup can be restored will allow swap, otherwise false.

backup_server_id - the ID of the backup server on which the backup is stored.

backup_size - the size of the backup

built - true if the backup is already built, otherwise false

built_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

data_store_type - data store type: lvm, vmware or solidfire

id - the backup ID

identifier - the backup identifier

initiated - period when backup is initiated: days, weeks, months, or years

iqn - volume ISCSI qualified name (SolidFire-related parameter)

locked - true if the backup is being built, otherwise false

marked_for_delete - the backup is marked for deletion (for auto-backups)
min_disk_size - minimum disk size required for restoring a backup

min_memory_size - minimum memory size required for restoring a backup

note - an optional note to the backup

operating_system - the OS of the VS from which the backup was created

operating_system_distro - the OS distribution of the VS from which the backup was created

target_id - ID of a backup target

target_type - target for which the backup was taken; For normal backups it is a disk. For incremental backups it's virtual server.

template_id - the ID of the template the VS is based on

updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

user_id - the ID of a user the storage server belongs to

volume_id - data store ID

backup_type - disk backup type (normal, Days auto-backup, Weeks auto-backup, Months auto-backup, Years auto-backup)

disk_id - the ID of the backed up disk

For details refer to Get The List Of Backup Servers section.

Note that the backup search returns only the list of backups, stored on a specified backup server, which a user has permission to see (own backups or all backups). For instructions on how to see the list of all backup servers, refer to Get The List of Backup Servers section.

19.9 Create CloudBoot Backup Server

To create a backup server where users will be able to store backups and templates, use the following request:

POST /settings/backup_servers.xml

POST /settings/backup_servers.json

Before creating a Cloud Boot backup server, you have to create new KVM compute resource with an IP address from the dynamic range.

XML Request Example

curl -I -X POST -u user: userpass
http://onapp.test/settings/backup_servers.xml -d
'<?xml version="1.0" encoding="UTF-8"?>
<backup_server><label>az_val</label><enabled>1</enabled><capacity>40</capacity><backup>true</backup><ip_address>172.0.0.1</ip_address><backup_ip_address>192.168.123.1</backup_ip_address></backup_server>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example
Where:

- **label** – backup server label
- **ip_address** – IP address of a KVM compute resource
- **backup** – set true to create a Cloud Boot backup server
- **backup_server_ip_address** – provisioning network IP address
- **capacity** – set the backup server capacity
- **enabled** – set the "enabled" as 1 if you want the backup server to be enabled, or 0 if you want it to be disabled. If you skip the **enabled** parameter, the backup server will be disabled by default.

Please note: You should configure some local or remote attached storage for persistent backups on the provisioning/backup server.

XML Output Example

```xml
<backup_server>
  <label>az_val_1</label>
  <created_at type="datetime">2012-02-10T15:14:53Z</created_at>
  <updated_at type="datetime">2012-02-10T15:14:53Z</updated_at>
  <backup_ip_address>192.168.123.1</backup_ip_address>
  <backup_server_group_id nil="true"/>
  <id type="integer">25</id>
  <enabled type="boolean">true</enabled>
  <backups type="array"/>
  <capacity type="integer">40</capacity>
  <ip_address>172.0.0.2</ip_address>
</backup_server>
```
20 Backup Server Zones

Backup server zone consists of several backup servers that share the same user permissions and are assigned to one billing plan. Backup server zones can be used for organizing and managing backup servers and creating different tiers of servers for customers.

20.1 Get List of Backup Server Zones

To get the list of backup server zones, use the following request:

GET /settings/backup_server_zones.xml
GET /settings/backup_server_zones.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<backup_server_groups type="array">
<backup_server_group>
  <label>bsz</label>
  <location_group_id type="integer">1</location_group_id>
  <created_at type="datetime">2012-01-04T11:50:40Z</created_at>
  <updated_at type="datetime">2012-01-04T11:50:40Z</updated_at>
  <id type="integer">28</id>
</backup_server_group>
</backup_server_groups>
```

Where:

- **label** – backup server zone title
- **location_group_id** – ID of a location group the backup server zone is assigned to
- **id** – backup server zone ID

Page History

v. 3.1

- added the location_group_id parameter

20.2 Get Backup Server Zone Details

To get the backup server zone details, use the following request:
GET /settings/backup_server_zones/:id.xml
GET /settings/backup_server_zones/:id.json

**XML Request Example**

curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url

**JSON Request Example**

curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url

**XML Output Example**

```xml
<backup_server_group>
  <label>bsz</label>
  <location_group_id type="integer">1</location_group_id>
  <created_at type="datetime">2012-01-04T11:50:40Z</created_at>
  <updated_at type="datetime">2012-01-04T11:50:40Z</updated_at>
  <id type="integer">28</id>
</backup_server_group>
```

*Where:*

- **label** – backup server zone title
- **location_group_id** – ID of a location group the backup server zone is assigned to
- **id** – backup server zone ID

**Page History**

v. 3.1

- added the `location_group_id` parameter

**20.3 Add Backup Server Zone**

To create a backup server zone, use the following request:

POST /settings/backup_server_zones.xml
POST /settings/backup_server_zones.json

**XML Request Example**

curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_server_zones.xml -d
"<backup_server_group><label>az_val_xml</label><location_group_id>1</location_group_id><backup_server_group>"

**JSON Request Example**

curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_server_zones.json -d
'{"backup_server_group":{"label":"az_val_json","location_group_id":"1"}}'
-H 'Accept: application/json' -H 'Content-type: application/json'

Where:

*label* – specify a new label of the backup server zone

*server_type* – specify the type of this backup zone, it can be either *virtual* or *smart*.

*location_group_id* – ID of a location group you wish to assign the backup server zone to

**Page History**

v. 5.3

- added the *server_type* parameter

v. 3.1

- added the *location_group_id* parameter

### 20.4 Edit Backup Server Zone

To edit backup server zone, use the following request:

PUT /settings/backup_server_zones/:id.xml
PUT /settings/backup_server_zones/:id.json

**XML Request Example**

curl -i -X PUT -u user:userpass
http://onapp.test/settings/backup_server_zones/2.xml -d
'<!--backup_server_group><label>az_val_change</label><location_group_id>1</location_group_id></backup_server_group-->' -H 'Accept: application/xml' -H
'Content-type: application/xml' -H

**JSON Request Example**

curl -i -X PUT -u user:userpass
http://onapp.test/settings/backup_server_zones/2.json -d
'{"backup_server_group":{"label":"az_val_change","location_group_id":"1"}}'
-H 'Accept: application/json' -H 'Content-type: application/json'

Where:

*label* – specify a new label of the backup server zone

*location_group_id* – ID of a location group you wish to assign the backup server zone to. You can change the already assigned location only if there are no backups or templates stores on backup servers of current zone.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no backup server zone with a requested ID, or URL is incorrect.
v. 3.1

- added the `location_group_id` parameter

## 20.5 Delete Backup Server Zone

To delete a backup server zone, use the following request:

DELETE /settings/backup_server_zones/:id.xml
DELETE /settings/backup_server_zones/:id.json

**XML Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/settings/backup_server_zones/2.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/settings/backup_server_zones/2.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no backup server zone with a requested ID, or URL is incorrect.

## 20.6 Get List of Servers Assigned to Backup Server Zone

To get the list of servers assigned to the backup server zone, use the following request:

GET /settings/backup_server_zones/:backup_server_zone_id/backup_servers.xml
GET /settings/backup_server_zones/:backup_server_zone_id/backup_servers.json

**XML Request Example**

```
curl -i -X GET -u user:userpass
http://onapp.test/settings/backup_server_zones/12/backup_servers.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X GET -u user:user:pass
http://onapp.test/settings/backup_server_zones/12/backup_servers.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```
**XML Output Example**

```xml
<backup_servers type="array">
  <backup_server>
    <label>az_val_ue_xml</label>
    <created_at type="datetime">2012-02-10T15:14:53Z</created_at>
    <updated_at type="datetime">2012-02-10T15:31:13Z</updated_at>
    <backup_server_group_id type="integer">55</backup_server_group_id>
    <id type="integer">25</id>
    <enabled type="boolean">true</enabled>
    <backups type="array"/>
    <capacity type="integer">40</capacity>
    <ip_address>172.0.0.2</ip_address>
  </backup_server>
</backup_servers>
```

Where:

- **backup_servers** – the array of backup servers assigned to this zone with the following details:
  - **label** – backup server label
  - **created at** – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
  - **updated at** – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
  - **backup_server_group_id** – the ID of the backup server zone the backup server belongs to
  - **id** – the backup server ID
  - **enabled** – backup server parameter; if "enabled" = true, the backup server is enabled; if "enabled"=false, the backup server is disabled.
  - **backups** – the list of backups stored at the backup server
  - **capacity** – the backup server capacity
  - **ip_address** – the backup server IP

### 20.7 Assign Backup Server to Backup Server Zone

To assign a backup server to a backup server zone, use the following request:

**POST**

```
http://onapp.test/settings/backup_server_zones/:backup_server_zone_id/backup_servers/:backup_server_id/attach.xml
```

Using this request you attach an unassigned backup server (:backup_server_id *) to a backup server zone (:backup_server_zone_id *).

When you add a backup server to a backup server zone, it inherits the zone's type. For more information refer to [Zone Types](#).

**XML Request Example**

```
curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_server_zones/2/backup_servers/12/attach.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```
JSON Request Example

```bash
curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_server_zones/2/backup_servers/12/attach.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

20.8 Unassign Backup Server from Backup Server Zone

To unassign a backup server from a backup server zone, use the following request:

POST /settings/backup_server_zones/backup_server_zone_id/backup_server/:backup_server_id/detach.xml

POST /settings/backup_server_zones/backup_server_zone_id/backup_server/:backup_server_id/detach.json

Using this request you detach an assigned backup server (:backup_server_id *) from a backup server zone (:backup_server_zone_id *)

XML Request Example

```bash
curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_server_zones/12/backup_servers/1/detach.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X POST -u user:userpass
http://onapp.test/settings/backup_server_zones/12/backup_servers/1/detach.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```
21 Baremetal Servers

Baremetal servers are physical servers deployed for a single user, that reside directly on the hardware without the virtualization layer.

NOTE: VLANs are not configured automatically on baremetal servers. You need to configure them manually in accordance with your OS and hardware settings.

21.1 Get List of Baremetal Servers

To get the list of all baremetal server in the cloud, use the following request:

GET /baremetal_servers.xml
GET /baremetal_servers.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example
<baremetal_servers type="array">
<baremetal_server>
<admin_note nil="true"/>
<allowed_swap type="boolean">true</allowed_swap>
<built type="boolean">true</built>
<cpu_sockets nil="true"/>
<cpu_threads nil="true"/>
<created_at type="datetime">2015-03-04T17:10:24+02:00</created_at>
<deleted_at nil="true"/>
<hostname>zaza</hostname>
<hypervisor_id type="integer">123</hypervisor_id>
?id type="integer">3842</id>
<identifier>vjktmvdlf0pyg3</identifier>
<initial_root_password>qwaszx!Q2</initial_root_password>
<initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
<label>zaza_BM</label>
<local_remote_access_ip_address nil="true"/>
<locked type="boolean">false</locked>
<note nil="true"/>
<operating_system>linux</operating_system>
<operating_system_distro>ubuntu</operating_system_distro>
<recovery_mode nil="true"/>
<state>delivered</state>
<template_id type="integer">19</template_id>
<template_label>debian-7.0-x64-1.4-xen.kvm.kvm_virtio.tar.gz</template_label>
(updated_at type="datetime">2015-03-04T17:15:26+02:00</updated_at>
<user_id type="integer">1</user_id>
<ip_addresses type="array">
<ip_address>
<address>109.123.105.156</address>
<broadcast>109.123.105.159</broadcast>
<created_at type="datetime">2014-01-15T11:18:12+02:00</created_at>
<disallowed_primary type="boolean">false</disallowed_primary>
<gateway>109.123.105.145</gateway>
<hypervisor_id nil="true"/>
?id type="integer">324</id>
<ip_address_pool_id nil="true"/>
<network_address>109.123.105.144</network_address>
<network_id type="integer">4</network_id>
<pxe type="boolean">false</pxe>
<updated_at type="datetime">2014-01-15T11:18:12+02:00</updated_at>
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>255.255.255.240</netmask>
</ip_address>
</ip_addresses>
</baremetal_server>
</baremetal_servers>

Where:

admin_note - an optional note of the administrator
allowed_swap - true if swap disk is allowed (depends on the template the server is based on); otherwise false
built - true if the server is built; otherwise false
cpu_sockets - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted
cpu_threads - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted.

cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in user billing plan.

created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

deleted_at - time when the VS was deleted

hostname - the name of your host

hypervisor_id - the ID of the compute resource used by this baremetal server

id - the baremetal server ID

identifier - the baremetal server identifier

initial_root_password - the baremetal server root password

initial_root_password_encrypted - true, if the baremetal server root password is encrypted, otherwise false

template_id - the ID of the template the baremetal server is based on

template_label - the name of the template from which this baremetal server is built

updated_at - the date when the baremetal server was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

ip_addresses - an array of IP addresses with their details assigned to this baremetal server:

- address – baremetal server IP
- broadcast – a logical address at which all devices connected to a multiple-access communications network are enabled to receive datagrams.
- created_at – time when IP address was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- customer_network_id - customer network ID
- disallowed_primary – true if not allowed to be used as primary (for baremetal server), otherwise false
- gateway - gateway address
- hypervisor_id - the ID of a compute resource the IP address is associated with
- id - the ID of the IP address
- ip_address_pool_id - ID of the IP address pool the IP address is associated with
- pxe - true, if this compute resource address can be used for cloudbooting a compute resource
- updated_at - time when IP address was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- user_id - the ID of the user this IP address is assigned to
- free – true if free, otherwise false
- `netmask` — netmask for the IP address

### 21.2 Get Baremetal Server Details

To get the details of a particular baremetal server, use the following request:

GET `/baremetal_servers/:id.xml`
GET `/baremetal_servers/:id.json`

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```xml
<server>
  <id>3842</id>
  <name>Server1</name>
  <hostname>server1.onapp.test</hostname>
  <image>example-image</image>
  <location>Your Location</location>
  <state>running</state>
  <status>success</status>
  <netmask>255.255.255.0</netmask>
  <ip_address>192.168.1.100</ip_address>
</server>
```
Where:

- **admin_note** - an optional note of the administrator
- **allowed_swap** - true if swap disk is allowed (depends on the template the server is based on); otherwise false
- **built** - true if the server is built; otherwise false
- **cpu_sockets** - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted
**cpu_threads** - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted.

**cpu_units** - the amount of CPU units per core if the CPU priority is replaced with CPU units in a bucket.

**created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

**deleted_at** - time when the VS was deleted

**hostname** - the name of your host

**hypervisor_id** - the ID of the compute resource used by this baremetal server

**id** - the baremetal server ID

**identifier** - the baremetal server identifier

**initial_root_password** - the baremetal server root password

**initial_root_password_encrypted** - true, if the baremetal server root password is encrypted, otherwise false

**label** - the baremetal server label

**local_remote_access_ip_address** - IP address used for remote access

**locked** - true if the baremetal server is locked; otherwise false

**note** - an optional reminder for this baremetal server made by a user account

**operating_system** - operating system used by the baremetal server

**operating_system_distro** - the distribution of the OS from which this baremetal server is built

**template_id** - the ID of the template the baremetal server is based on

**template_label** - the name of the template from which this baremetal server is built

**updated_at** - the date when the baremetal server was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**ip_addresses** - an array of IP addresses with their details assigned to this baremetal server:

- **address** – baremetal server IP
- **broadcast** – a logical address at which all devices connected to a multiple-access communications network are enabled to receive datagrams.
- **created_at** – time when the IP address was created in the [YYYY][MM][DD][ hh][mm][ss]Z format
- **customer_network_id** - customer network ID
- **disallowed_primary** – true if not allowed to be used as primary (for baremetal server), otherwise false
- **gateway** - gateway address
- **hypervisor_id** - the ID of a compute resource the IP address is associated with
- **id** – the ID of the IP address
- **ip_address_pool_id** - ID of the IP address pool the IP address is associated with
- **pxe** - true, if this compute resource address can be used for cloudbooting a compute resource
- **updated_at** - time when the IP address was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **user_id** - the ID of the user this IP address is assigned to
- **free** – true if free, otherwise false
• netmask — netmask for the IP address

21.3 Create Baremetal Server

The management network should be disconnected during the bare metal server deployment.

To create a baremetal server, use the following request:

POST /baremetal_servers.xml
POST /baremetal_servers.json

**XML Request Example**

```bash
curl -i -X POST http://onapp.test/baremetal_servers.xml -d '  <baremetal_server><template_id>2</template_id><label>test_baremetal</label><hostname>test</hostname><domain>localdomain</domain><hypervisor_group_id>121</hypervisor_group_id><hypervisor_id>38</hypervisor_id><initial_root_password>qwaszx</initial_root_password><primary_network_group_id>120</primary_network_group_id><selected_ip_address>5.1.1.12</selected_ip_address><required_ip_address_assignment>1</required_ip_address_assignment><recipe_ids type="array"><recipe_id>11</recipe_id></recipe_ids></baremetal_server>
```

**JSON Request Example**

```bash
curl -i -X POST http://onapp.test/baremetal_servers.json -d '{  "baremetal_server":{"template_id":"2","label":"test_baremetal","hostname":"test","domain":"localdomain","hypervisor_group_id":"121","hypervisor_id":"38","initial_root_password":"qwaszx","primary_network_group_id":"120","selected_ip_address":"5.1.1.12","required_ip_address_assignment":"1","recipe_ids:["11"]}}' -u user:password -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**Where:**

*template_id* - the ID of a template from which a baremetal server should be built

*label* - user-friendly baremetal server description

*hostname* - specify the baremetal server hostname

*domain* - specify the domain for this VS. The default value is *localdomain*. This parameter is not applicable for Windows virtual servers.

*hypervisor_group_id* - the ID of the baremetal zone in which the server will be created. Optional: if no compute zone is set, the baremetal server will be built in any available baremetal compute zone.

*hypervisor_id* - the ID of a baremetal compute resource where the baremetal server will be built. If no baremetal compute resource ID is specified, the server will be built on the compute resource with the least available RAM (but sufficient RAM for the server)
**initial_root_password** - the root password for a baremetal server. If none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-], and lower dash [ _ ], and the following special characters: ~ ! @ # $ * _ + = \ { } [ ] : ; ' , . ? /. You can use both lower- and uppercase letters.

**primary_network_group_id** - the ID of the primary network group. Optional parameter.

**required_ip_address_assignment** - set "1" if you want IP address to be assigned automatically after creation. Otherwise set "0"

**selected_ip_address** - an IP address to assign to this VS; if the above parameter **required_ip_address_assignment** was set "1" but this parameter **selected_ip_address** is empty - the first available IP address will be assigned to VS automatically

**recipe_ids** - an array of recipe IDs that can be used during the recipe server creation. You can only run recipes on baremetal server provisioning.

---

**Page History**

v.5.4

- added the following parameters:
  - domain
  - selected_ip_address
- removed selected_ip_address_id parameter

---

### 21.4 Delete Baremetal Server

To delete a baremetal server, use the following request:

DELETE /baremetal_servers/:id.xml

DELETE /baremetal_servers/:id.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass
```

**Where:**

*id* – the ID of a baremetal server you want to delete

---

### 21.5 Add/Edit Admin/User Note for Baremetal Server

To edit/make an admin note, use the following request:

PUT /baremetal_servers/:baremetal_server_id.xml

PUT /baremetal_servers/:baremetal_server_id.json
XML Request Example

```bash
```

JSON Request Example

```bash
```

Where:

- **admin_note** – enter the text of your note.
- **baremetal_server_id** - the ID of the baremetal server for which you add/edit a note.

To edit/make a user note, use the following request:

XML Request Example

```bash
```

JSON Request Example

```bash
```

Where:

- **note** – enter the text of your note.
- **baremetal_server_id** - the ID of the baremetal server for which you add/edit a note.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no baremetal server with a requested ID, or URL is incorrect.

### 21.6 Enable Recovery Mode for Baremetal Server

To enable recovery mode for baremetal server, use the following request:

- **PUT** /baremetal_servers/:id/enable_recovery.xml
- **PUT** /baremetal_servers/:id/enable_recovery.json

XML Request Example

```bash
```
21.7 Disable Recovery Mode for Baremetal Server

To disable recovery mode for baremetal server, use the following request:

**PUT /baremetal_servers/:id/enable_recovery.xml**

**PUT /baremetal_servers/:id/enable_recovery.json**

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```
22 Buckets

Buckets define the resource allocation and prices for resources in the cloud. They are made up of two parts:

- Access Control which defines the resources the user under the bucket has access to
- Rate Card which includes the prices for resource usage

This section contains the API requests you can use to manage buckets.

22.1 Get List of Buckets

To get the list of buckets, use the following request:

GET /billing/buckets.xml
GET /billing/buckets.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<buckets type="array">
  <bucket>
    <id type="integer">3</id>
    <label>ut</label>
    <created_at type="dateTime">2017-06-26T08:48:05+00:00</created_at>
    <updated_at type="dateTime">2017-06-26T08:48:05+00:00</updated_at>
    <currency_code>USD</currency_code>
    <show_price nil="true"/>
    <monthly_price type="decimal">1.0</monthly_price>
    <allows_mak type="boolean">true</allows_mak>
    <allows_kms type="boolean">true</allows_kms>
    <allows_own type="boolean">true</allows_own>
    <associated_with_users type="integer">1</associated_with_users>
  </bucket>
</buckets>
```

Where:

- **id** - the bucket ID
- **label** - the bucket name
- **created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
updated_at - the date when the bucket was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

currency_code - the currency in which the users are charged

show_price - true, if users can see the prices set up for them, otherwise false

monthly_price * - the monthly fee for bucket usage

allows_mak - true, if the MAK licensing is allowed, otherwise false

allows_kms - true, if the KMS licensing is allowed for this bucket, otherwise false

allows_own - true, if adding own licenses is allowed for this bucket, otherwise false

associated_with_users - the number of users with which this bucket is associated

22.2 Get Bucket Details

To get bucket details, use the following request:

GET /billing/buckets/id.xml
GET /billing/buckets/id.json

XML Request Example


JSON Request Example


XML Output Example

```xml
<bucket>
  <id type="integer">21</id>
  <label>minima</label>
  <created_at type="dateTime">2017-06-26T08:48:09+00:00</created_at>
  <updated_at type="dateTime">2017-06-26T08:48:09+00:00</updated_at>
  <currency_code>USD</currency_code>
  <show_price nil="true"/>
  <monthly_price type="decimal">1.0</monthly_price>
  <allows_mak type="boolean">true</allows_mak>
  <allows_kms type="boolean">true</allows_kms>
  <allows_own type="boolean">true</allows_own>
  <type>Billing::Buckets::Plan</type>
  <associated_with_users type="integer">1</associated_with_users>
</bucket>
```

Where:

id - the bucket ID

label - the bucket name
created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
updated_at - the date when the bucket was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
currency_code - the currency in which the users are charged
show_price - true, if users can see the prices set up for them, otherwise false
monthly_price * - the monthly fee for bucket usage
allows_mak - true, if the MAK licensing is allowed, otherwise, false
allows_kms - true, if the KMS licensing is allowed for this bucket, otherwise, false
allows_own - true, if adding own licenses is allowed for this bucket, otherwise, false
type - the type of bucket
associated_with_users - the number of users with which this bucket is associated

22.3 Add Bucket

To create a new bucket, use the following request:

POST /billing/buckets.xml
POST /billing/buckets.json

XML Request Example

```
curl -i -X POST http://onapp.test/billing/buckets.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml' -u user:userpass -d '<bucket><label>Label</label><currency_code>USD</currency_code><monthly_price type="integer">10</monthly_price><allows_kms type="boolean">false</allows_kms><allows_mak type="boolean">true</allows_mak><allows_own type="boolean">false</allows_own></bucket>'
```

JSON Request Example

```
```

Where:

- label * - the bucket name
- currency_code * - the currency that users will be charged in within this bucket (USD by default)
- monthly_price * - set the monthly fee for bucket usage
- allows_kms - true, if the KMS licensing is allowed for this bucket, otherwise, false
- allows_mak - true, if the MAK licensing is allowed, otherwise, false
- allows_own - true, if adding own licenses is allowed for this bucket, otherwise, false

22.4 Edit Bucket

To edit a bucket, use the following request:
PUT /billing/buckets/:bucket_id.xml
PUT /billing/buckets/:bucket_id.json

XML Request Example

```bash
  <bucket>
    <label>NewLabel</label>
    <monthly_price type="integer">22</monthly_price>
  </bucket>'
```

JSON Request Example

```bash
  {
    "label": "NewLabel",
    "monthly_price": 22
  }'
```

Where:

- `label` - the bucket name
- `monthly_price` - set the monthly fee for bucket usage

### 22.5 Delete Bucket

To delete a bucket, use the following request:

DELETE /billing/buckets/id.xml
DELETE /billing/buckets/id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

### 22.6 Clone Bucket

To clone a bucket with its prices and added resources, use the following request:

POST /billing/buckets/:bucket_id/clone.xml
POST /billing/buckets/:bucket_id/clone.json

XML Request Example

JSON Request Example


Where you indicate in the URL the ID of the cloned bucket.

22.7 Access Control

The Access Control is the part of the bucket which defines the resources to which a user under the bucket has access. In the Access Control, you define the maximum/minimum/default limits for resource usage. If a resource is not added to the Access Control a user under the bucket will not have access to that resource. This section contains the API requests you can use to manage Access Controls.

22.7.1 Get List of Access Controls for Baremetal Server Type

To get the list of access controls for baremetal server type, use the following request:
GET /billing/buckets/:bucket_id/access_controls.xml
GET /billing/buckets/:bucket_id/access_controls.json

XML Request Example


JSON Request Example


XML Output Example
Where:

- **bucket_id** - the ID of the bucket with which this access control is associated
- **server_type** - baremetal server type
- **target_id** - the ID of the zone that is added to this access control
- **type** - the type of the resource which is added to a bucket, it can be one of the following values:
  - `compute_zone_resource`
  - `bare_metal_servers_resource`
  - `network_zone_resource`
- **timing_strategy** - the type of billing for each resource: hourly for baremetal server type
- **preferences** - this parameter is empty for baremetal server type
- **limits** - the array of limits for the resource. Depending on the type of resource, you will have the following parameters:
  - for the `bare_metal_servers_resource` resource: **limit** - the total amount of baremetal servers allowed
  - for the `network_zone_resource` resource: **limit_ip** - the total amount of IP addresses

### 22.7.2 Get List of Access Controls for Smart Server Type

To get the list of access controls, use the following request:

GET /billing/buckets/:bucket_id/access_controls.xml
GET /billing/buckets/:bucket_id/access_controls.json

**XML Request Example**

```
curl -X GET http://onapp.test/billing/buckets/5/access_controls.xml -u user:userpass
```

**JSON Request Example**

```
curl -X GET http://onapp.test/billing/buckets/5/access_controls.json -u user:userpass
```

**XML Output Example**

```xml
<access_controls type="array">
  <access_control>
    <bucket_id>34</bucket_id>
    <server_type>baremetal</server_type>
    <type>bare_metal_servers_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <preferences/>
    <limits>
      <limit>12.0</limit>
    </limits>
  </access_control>
</access_controls>
```
<access_controls type="array">
  <access_control>
    <bucket_id type="integer">5</bucket_id>
    <server_type>smart</server_type>
    <target_id type="integer">6</target_id>
    <type>backup_server_zone_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>BackupSmartZone</target_name>
    <preferences/>
    <limits>
      <limit_backup type="decimal">10.0</limit_backup>
      <limit_backup_disk_size type="decimal">150.0</limit_backup_disk_size>
      <limit_template type="decimal">10.0</limit_template>
      <limit_template_disk_size type="decimal">10.0</limit_template_disk_size>
    </limits>
  </access_control>
  ...
</access_controls>

Where:

bucket_id - the ID of the bucket with which this access control is associated

server_type - smart server type

target_id - the ID of the zone that is added to the access control

type - the type of the resource that is added to the bucket, it can be one of the following values:

- backups_resource
- backup_server_zone_resource
- compute_resource_storing_resource
- compute_zone_resource
- data_store_zone_resource
- network_zone_resource
- smart_servers_resource

timing_strategy - the type of billing for each resource (hourly for Smart server type)

target_name - the name of the resource that is added to the access control. For example, it can be a label of a network or compute zone, etc.

preferences - this parameter does not apply to smart server type.

limits - the array of limits for the resource

### Smart Server Limits

<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backups_resource</td>
<td>limit</td>
<td>the maximum number of backups stored on a compute resource users can create (backup/hour)</td>
</tr>
<tr>
<td>Smart Server Limits</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>backup_server_zone_resource limit_backup</td>
<td>the maximum amount of backups users can create in this backup server zone under the bucket (backup/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_backup_disk_size</td>
<td>the maximum amount of disk space users get for storing their backups in this backup server zone under the bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_template</td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_template_disk_size</td>
<td>the maximum amount of disk space users get for storing their templates in this backup server zone under the bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>compute_resource_storing_resource limit</td>
<td>the total amount of disk space users can request for storing their backups, ISOs and templates under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>compute_zone_resource</td>
<td></td>
<td></td>
</tr>
<tr>
<td>limit_cpu</td>
<td>the maximum number of CPU cores that can be set in the smart server creation wizard when the user adds a server under this bucket in the compute zone (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_cpu_share</td>
<td>the maximum amount of CPU shares users can request for all their servers in this compute zone under this bucket (CPU share %/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_cpu_units</td>
<td>the maximum amount of CPU units that users can request for all their smart servers within this compute zone under the bucket (CPU unit/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_memory</td>
<td>the maximum amount of RAM that users can request for all their smart servers within this compute zone under the bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>use_cpu_units</td>
<td>Set to &quot;1&quot; to use CPU units instead of CPU shares (CPU unit/hour), otherwise, set to &quot;0&quot;. Set the amount of CPU units available to users under this bucket using the limit_cpu_units parameter.</td>
<td></td>
</tr>
</tbody>
</table>
22.7.3 Get List of Access Controls for Virtual Server Type

To get the list of access controls, use the following request:

GET /billing/buckets/:bucket_id/access_controls.xml
GET /billing/buckets/:bucket_id/access_controls.json

XML Request Example

curl "http://onapp.test/billing/buckets/10/access_controls.xml" -X GET \ -u user:userpass

JSON Request Example

curl "http://onapp.test/billing/buckets/10/access_controls.json" -X GET \ -u user:userpass

XML Output Example
<access_controls type="array">
  <access_control>
    <bucket_id type="integer">10</bucket_id>
    <server_type>virtual</server_type>
    <target_id type="integer">7</target_id>
    <type>backup_server_zone_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>BackupServerZone</target_name>
    <preferences/>
    <limits>
      <limit_backup type="decimal">10.0</limit_backup>
      <limit_backup_disk_size type="decimal">150.0</limit_backup_disk_size>
      <limit_template type="decimal">10.0</limit_template>
      <limit_template_disk_size type="decimal">10.0</limit_template_disk_size>
    </limits>
  </access_control>
</access_controls>

Where:

- **bucket_id** - the ID of the bucket with which this access control is associated
- **server_type** - virtual server type
- **target_id** - the ID of the zone that is added to the access control
- **type** - the type of the resource that is added to the bucket, it can be one of the following values:
  - `network_zone_resource`
  - `compute_zone_resource`
  - `backup_server_zone_resource`
  - `solidfire_data_store_zone_resource`
  - `virtual_servers_resource`
  - `autoscaled_servers_resource`
  - `templates_resource`
  - `compute_resource_storing_resource`
  - `backups_resource`
  - `iso_templates_resource`
  - `application_servers_resource`
  - `container_servers_resource`
- `data_store_zone_resource`

- `preconfigured_servers_resource`

参加方式 - 每个资源的计费类型：hourly or monthly (on peak usage)

- `target_name` - 资源名称，用于访问控制。例如，它可以是一个网络或计算区的标签等。

- `preferences` - 区域ID的数组，用于用于预配置服务器资源类型。

`limits` - 资源的限制数组。

<table>
<thead>
<tr>
<th>Virtual Server Preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
</tr>
<tr>
<td><code>preconfigured_servers_resource</code></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Virtual Server Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>network_zone_resource</code></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

| `compute_zone_resource` | `limit_cpu` | the maximum amount of CPU cores that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU core/hour) |
|                       | `limit_cpu_share` | the maximum amount of CPU shares users can request for all their servers in this compute zone under this bucket (CPU share %/hour) |
|                       | `limit_cpu_units` | the maximum amount of CPU units that users can request for all their VSs within this compute zone under the bucket (CPU unit/hour) |
|                       | `limit_memory` | the maximum amount of RAM that users can request for all their VSs within this compute zone under the bucket (GB/hour) |
|                       | `limit_default_cpu` | the default amount of CPU cores that will be set in the VS creation wizard when the user adds a VS in
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>limit_min_cpu</td>
<td>the minimum amount of CPU cores that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU core/hour)</td>
</tr>
<tr>
<td>limit_min_memory</td>
<td>the minimum amount of RAM that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (GB/hour)</td>
</tr>
<tr>
<td>limit_default_cpu_share</td>
<td>the default amount of CPU shares that will be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU share %/hour)</td>
</tr>
<tr>
<td>limit_min_cpu_priority</td>
<td>the minimum amount of CPU priority which can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU share %/hour)</td>
</tr>
<tr>
<td>use_cpu_units</td>
<td>set to &quot;1&quot; to use CPU shares instead of CPU priority (CPU unit/hour). Otherwise, set to &quot;0&quot;. Set the amount of CPU units available to users under this bucket using the limit_cpu_units parameter.</td>
</tr>
<tr>
<td>use_default_cpu</td>
<td>set to &quot;1&quot; if a default amount of CPU cores is to be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU core/hour). Otherwise, set to &quot;0&quot;. Set the default amount of CPU cores using the limit_default_cpu parameter.</td>
</tr>
<tr>
<td>use_default_cpu_share</td>
<td>set to &quot;1&quot; if a default amount of CPU shares is to be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU share %/hour). Otherwise, set to &quot;0&quot;. Set the default amount of CPU shares using the limit_default_cpu_share parameter.</td>
</tr>
<tr>
<td>backup_server_zone_resource</td>
<td>limit_backup</td>
</tr>
<tr>
<td></td>
<td>the maximum amount of backups users can create in this backup server zone under the bucket (backup/hour)</td>
</tr>
</tbody>
</table>
## Virtual Server Preferences

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>limit_backup_disk_size</code></td>
<td>the maximum amount of disk space users get for storing their backups in this backup server zone under the bucket</td>
</tr>
<tr>
<td><code>limit_template</code></td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td><code>limit_template_disk_size</code></td>
<td>the maximum amount of disk space users get for storing their templates in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td><code>limit_ova</code></td>
<td>the maximum amount of OVAs users can create in this backup server zone under the bucket (OVA/hour)</td>
</tr>
<tr>
<td><code>limit_ova_disk_size</code></td>
<td>the maximum amount of disk space users get for storing their OVAs in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td><code>solidfire_data_store_zone_resource</code></td>
<td>the maximum number of IOPS available under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td><code>virtual_servers_resource</code></td>
<td>the maximum number of virtual servers users can create in the cloud. This parameter affects the number of virtual servers, VSs in Federation and load balancers users can create (VS/hour)</td>
</tr>
<tr>
<td><code>autoscaled_servers_resource</code></td>
<td>the maximum number of VSs for which the user can enable autoscaling under this bucket (VS/hour)</td>
</tr>
<tr>
<td><code>templates_resource</code></td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td><code>compute_resource_storing_resource</code></td>
<td>the total amount of disk space users can request for storing their backups, ISOs and templates under this bucket (GB/hour)</td>
</tr>
<tr>
<td><code>backups_resource</code></td>
<td>the maximum number of backups users can create under this bucket (backup/hour)</td>
</tr>
<tr>
<td><code>iso_templates_resource</code></td>
<td>the maximum number of ISO templates users can create under this bucket. (ISO/hour)</td>
</tr>
</tbody>
</table>

177
Virtual Server Preferences

<table>
<thead>
<tr>
<th>resource</th>
<th>limit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>application_servers_resource</td>
<td>limit</td>
<td>the maximum number of application servers in the cloud that the users can create under this bucket. (application VS/hour)</td>
</tr>
<tr>
<td>container_servers_resource</td>
<td>limit</td>
<td>the maximum number of container servers in the cloud that the users can create under this bucket. (container VS/hour)</td>
</tr>
<tr>
<td>data_store_zone_resource</td>
<td>limit</td>
<td>the maximum amount of disk space (GB) users can request in the data store zone under the bucket. (GB/hour)</td>
</tr>
</tbody>
</table>

Page History

v.6.0
- removed the accelerated_servers_resource parameter

v.5.7
- added the preferences parameter
- added the preconfigured_servers_resource resource type that can have the following parameters:
  - hypervisor_group_ids
  - data_store_group_ids
  - network_group_ids
- removed the legacy_resource_id parameter

22.7.4 Get List of Access Controls for Other Server Type

To get the list of access controls, use the following request:

GET /billing/buckets/:bucket_id/access_controls.xml
GET /billing/buckets/:bucket_id/access_controls.json

XML Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/billing/buckets/:bucket_id/access_controls.xml -H
'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/billing/buckets/:bucket_id/access_controls.json -H
'Accept: application/json' -H 'Content-type: application/json'
```
XML Output Example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<access_controls type="array">
  <access_control>
    <bucket_id type="integer">5263</bucket_id>
    <server_type>other</server_type>
    <target_id type="integer">14</target_id>
    <type>template_groups_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>template_name</target_name>
    <preferences>
    </preferences>
    <limits>
    </limits>
  </access_control>
  <access_control>
  </access_control>
</access_controls>
```

Where:

- **bucket_id** - the ID of the bucket with which this access control is associated
- **server_type** - *other* server type
- **target_id** - the ID of the zone for which the limits are set.
- **type** - the type of the resource that is added to a bucket, it can be one of the following values:
  - `backup_resource_zone_resource`
  - `template_resource`
  - `edge_groups_resource`
  - `cdn_bandwidth_resource`
  - `recipe_groups_resource`
  - `service_addon_resource`
  - `blueprint_groups_resource`

- **timing_strategy** - the type of billing for each resource.
- **target_name** - the label of the resource added to the access control
- **preferences** - this parameter does not apply to *other* server types.
- **limits** - the array of limits for the resource. The access control for *Other* resources includes the `cdn_bandwidth_resource` limit allocated in GB per month.

22.7.4.1 Page History

v. 6.3 Edge 1

Added the `blueprint_groups_resource` resource type.

v. 5.8

Added the following resource types:
- `backup_resource_zone_resource`
- `cdn_bandwidth_resource`

22.7.5 Add Access Control for Baremetal Server Type

To create a new access control, use the following request:
POST /billing/buckets/:bucket_id/access_controls.xml
POST /billing/buckets/:bucket_id/access_controls.json

XML Request Example

```
curl "http://onapp.test/billing/buckets/34/access_controls.xml"
<access_control><bucket_id>34</bucket_id><server_type>baremetal</server_type><preferences><limits><limit>12.0</limit></limits></preferences></access_control>" -X POST -u user:userpass -H "Accept: application/xml" -H "Content-Type: application/xml"
```

JSON Request Example

```
curl http://onapp.test/billing/buckets/34/access_controls.json -d '{"access_control": {"bucket_id": 34, "server_type": "baremetal", 
"target_id":null, "create_rate_card": false, "type": "bare_metal_servers_resource", "limits": {"limit": 12}}}'}-X POST -u user:userpass -H "Accept: application/json" -H "Content-Type: application/json"
```

Where:

- `bucket_id` - the ID of the bucket with which this access control is associated
- `server_type` - the server type this access control is applicable to. Can be virtual, smart, baremetal or vpc.
- `target_id` - the ID of the resource that is added to the access control
- `type` - the type of the resource that is added to the access control, it can be one of the following values:
  - `bare_metal_servers_resource`
  - `compute_zone_resource`
  - `network_zone_resource`
- `timing_strategy` - the type of billing for each resource: hourly or monthly (on peak usage).
- `limits` - the array of limits for a resource. Depending on the type of resource, you will have the following parameters:
  - for the `bare_metal_servers_resource` resource: `limit` - the total amount of baremetal servers available under this bucket
  - for the `compute_zone_resource` resource: `target_id` - a label of a compute zone that you want to add to access control
  - for the `network_zone_resource` resource: `limit_ip` - the total amount of IP addresses available under this bucket

Page History

- v.6.2 Edge 1
  - removed the `apply_to_all_resources_in_the_bucket` parameter
- v.5.9
  - added the `apply_to_all_resources_in_the_bucket` parameter
22.7.6 Add Access Control for Smart Server Type

To create a new access control for smart server type, use the following request:

POST /billing/buckets/:bucket_id/access_controls.xml
POST /billing/buckets/:bucket_id/access_controls.json

XML Request Example

```
curl -i -X POST http://onapp.test/billing/buckets/5/access_controls.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml' -u user:pass -d '<access-control><bucket_id type="integer">5</bucket_id><server_type>smart</server_type><target_id type="integer">6</target_id><timing_strategy>hourly</timing_strategy><target_name>BackupSmartZone</target_name><limits><limit_backup type="decimal">10.0</limit_backup><limit_backup_disk_size type="decimal">150.0</limit_backup_disk_size><limit_template type="decimal">10.0</limit_template_disk_size><limit_template_disk_size type="decimal">10.0</limit_template_disk_size></limits></access_control>'
```

JSON Request Example

```
curl -i -X POST http://onapp.test/billing/buckets/5/access_controls.json -H 'Accept: application/json' -H 'Content-Type: application/json' -u user:pass -d '{"bucket_id": 5, "server_type": "smart", "target_id": 6, "type": "backup_server_zone_resource", "timing_strategy": "hourly", "target_name": "BackupSmartZone", "limits": {"limit_backup": 10.0, "limit_backup_disk_size": 150.0, "limit_template": 10.0, "limit_template_disk_size": 10.0}}'
```

Where:

- `bucket_id` * - the ID of the bucket with which this access control is associated
- `server_type` * - the server type this access control is applicable to (smart for Smart server type)
- `target_id` - the ID of the resource that is added to the access control
- `type` * - the type of the resource that is added to the access control, it can be one of the following values:
  - `backups_resource`
  - `backup_server_zone_resource`
  - `compute_resource_storing_resource`
  - `compute_zone_resource`
  - `data_store_zone_resource`
  - `network_zone_resource`
  - `smart_servers_resource`
- `timing_strategy` - the type of billing for each resource (hourly for Smart server type)
- `target_name` - the name of the resource that is added to the access control. For example, it can be a label of a network or compute zone, etc.
- `limits` - the array of limits for the resource
## Smart Server Limits

<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backups_resource</td>
<td>limit</td>
<td>the maximum number of backups users can create under this bucket (backup/hour)</td>
</tr>
<tr>
<td>backup_server_zone_resource</td>
<td>limit_backup</td>
<td>the maximum amount of backups users can create in this backup server zone under the bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_backup_disk_size</td>
<td>the maximum amount of disk space users get for storing their backups in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_template</td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_template_disk_size</td>
<td>the maximum amount of disk space users get for storing their templates in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td>compute_resource_storing_resource</td>
<td>limit</td>
<td>the total amount of disk space users can request for storing their backups, ISOs and templates under this bucket (GB/hour)</td>
</tr>
<tr>
<td>compute_zone_resource</td>
<td>limit_cpu</td>
<td>the maximum number of CPU cores that can be set in the smart server creation wizard when the user adds a server under this bucket in the compute zone (CPU core/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_cpu_share</td>
<td>the maximum amount of CPU shares users can request for all their servers in this compute zone under this bucket (CPU share %/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_cpu_units</td>
<td>the maximum amount of CPU units that users can request for all their smart servers within this compute zone under the bucket (CPU unit/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_memory</td>
<td>the maximum amount of RAM that users can request for all their smart servers within this compute zone under the bucket (GB/hour)</td>
</tr>
</tbody>
</table>
### Smart Server Limits

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>use_cpu_units</td>
<td></td>
<td>Set to &quot;1&quot; to use CPU units instead of CPU shares (CPU unit/hour), otherwise, set to &quot;0&quot;. Set the amount of CPU units available to users under this bucket using the <code>limit_cpu_units</code> parameter.</td>
</tr>
<tr>
<td>data_store_zone_resource</td>
<td>limit</td>
<td>the maximum amount of disk space (GB) users can request in the data store zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td>network_zone_resource</td>
<td>limit_ip</td>
<td>the maximum number of IP addresses users can request under this bucket (IP/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_rate</td>
<td>the maximum port speed user can request in this network zone under the bucket (Mbps/hour)</td>
</tr>
<tr>
<td>smart_servers_resource</td>
<td>limit</td>
<td>the maximum number of smart servers users can create in the cloud</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This parameter affects the number of smart servers, smart servers in Federation and load balancers users can create (VS/hour).</td>
</tr>
</tbody>
</table>

**Page History**

v.6.2 Edge 1
- removed the `apply_to_all_resources_in_the_bucket` parameter

v.5.9
- added the `apply_to_all_resources_in_the_bucket` parameter

**22.7.7 Add Access Control for Virtual Server Type**

To create a new access control, use the following request:

```plaintext```
POST /billing/buckets/:bucket_id/access_controls.xml
POST /billing/buckets/:bucket_id/access_controls.json
```

XML Request Example
curl "http://onapp.test/billing/buckets/5/access_controls.xml" -d '
  <access-control><bucket_id type="integer">5</bucket_id><server_type><target_id type="integer">6</target_id><backup_server_zone_resource><target_name>BackupServerZone</target_name><preferences><preferences/><limits><limit_backup type="decimal">10.0</limit_backup><limit_backup_disk_size type="decimal">150.0</limit_backup_disk_size><limit_template type="decimal">10.0</limit_template_disk_size><limit_template_disk_size type="decimal">10.0</limit_template_disk_size></limits></access_control>'
-X POST \ -u user:userpass \ -H "Accept: application/xml" \ -H "Content-Type: application/xml"

JSON Request Example

curl "http://onapp.test/billing/buckets/5/access_controls.json" -d '
  {"bucket_id": 5, "server_type": "virtual", "target_id": 6, "type": "backup_server_zone_resource", "timing_strategy": "hourly", "target_name": "BackupServerZone", "limits": {"limit_backup": 10.0, "limit_backup_disk_size": 150.0, "limit_template": 10.0, "limit_template_disk_size": 10.0}}' -X POST \ -u user:userpass \ -H "Accept: application/json" \ -H "Content-Type: application/json"

Where:

- **bucket_id** - the ID of the bucket with which this access control is associated
- **server_type** - the server type this access control is applicable to (virtual for Virtual server type)
- **target_id** - the ID of the resource that is added to the access control
- **type** - the type of the resource that is added to the access control, it can be one of the following values:
  - network_zone_resource
  - compute_zone_resource
  - backup_server_zone_resource
  - solidfire_data_store_zone_resource
  - virtual_servers_resource
  - autoscaled_servers_resource
  - templates_resource
  - compute_resource_storing_resource
  - backups_resource
- `isoTemplates_resource`
- `application_servers_resource`
- `container_servers_resource`
- `dataStore_zone_resource`
- `preconfigured_servers_resource`

**Timing strategy** - the type of billing for each resource: *hourly* or *monthly* (on peak usage)

**Target name** - the name of the resource that is added to the access control. For example, it can be a label of a network or compute zone, etc.

**Preferences** - the array of zone IDs added to an instance package. This parameter is available for the `preconfigured_servers_resource` resource type.

**Limits** - the array of limits for the resource

<table>
<thead>
<tr>
<th>Virtual Server Preferences</th>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>preconfigured_servers_resource</code></td>
<td><code>hypervisor_group_ids</code></td>
<td>The ID(s) of a compute zone added to an instance package.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>data_store_group_ids</code></td>
<td>The ID(s) of a data store zone added to an instance package.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>network_group_ids</code></td>
<td>The ID(s) of a network zone added to an instance package.</td>
</tr>
</tbody>
</table>

| Virtual Server Limits      | `network_zone_resource`       | `limit_ip`                        | the maximum number of IP addresses users can request under this bucket (IP/hour) |
|                            |                               | `limit_rate`                      | the maximum port speed amount user can request in this network zone under the bucket (Mbps/hour) |

|                            | `compute_zone_resource`       | `limit_cpu`                       | the maximum amount of CPU cores that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU core/hour) |
|                            |                               | `limit_cpu_share`                 | the maximum amount of CPU shares users can request for all their servers in this compute zone under this bucket (CPU share %/hour) |
|                            |                               | `limit_cpu_units`                 | the maximum amount of CPU units that users can request for all their VSs within this compute zone under the bucket (CPU unit/hour) |
### Virtual Server Preferences

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>limit_memory</code></td>
<td>The maximum amount of RAM that users can request for all their VSs within this compute zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td><code>limit_default_cpu</code></td>
<td>The default amount of CPU cores that will be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td><code>limit_min_cpu</code></td>
<td>The minimum amount of CPU cores that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU core/hour)</td>
</tr>
<tr>
<td><code>limit_min_memory</code></td>
<td>The minimum amount of RAM that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (GB/hour)</td>
</tr>
<tr>
<td><code>limit_default_cpu_share</code></td>
<td>The default amount of CPU shares that will be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU share %/hour)</td>
</tr>
<tr>
<td><code>limit_min_cpu_priority</code></td>
<td>The minimum amount of CPU priority which can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU share %/hour)</td>
</tr>
<tr>
<td><code>use_cpu_units</code></td>
<td>Set to &quot;1&quot; to use CPU shares instead of CPU priority (CPU unit/hour). Otherwise, set to &quot;0&quot;. Set the amount of CPU units available to users under this bucket using the <code>limit_cpu_units</code> parameter.</td>
</tr>
<tr>
<td><code>use_default_cpu</code></td>
<td>Set to &quot;1&quot; if a default amount of CPU cores is to be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU core/hour). Otherwise, set to &quot;0&quot;. Set the default amount of CPU cores using the <code>limit_default_cpu</code> parameter.</td>
</tr>
<tr>
<td><code>use_default_cpu_share</code></td>
<td>Set to &quot;1&quot; if a default amount of CPU shares is to be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU share %/hour). Otherwise, set to &quot;0&quot;. Set the default amount of CPU shares using</td>
</tr>
</tbody>
</table>
### Virtual Server Preferences

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Limit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>backup_server_zone_resource</code></td>
<td><code>limit_backup</code></td>
<td>the maximum amount of backups users can create in this backup server zone under the bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_backup_disk_size</code></td>
<td>the maximum amount of disk space users get for storing their backups in this backup server zone under the bucket</td>
</tr>
<tr>
<td></td>
<td><code>limit_template</code></td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_template_disk_size</code></td>
<td>the maximum amount of disk space users get for storing their templates in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_ova</code></td>
<td>the maximum amount of OVAs users can create in this backup server zone under the bucket (OVA/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_ova_disk_size</code></td>
<td>the maximum amount of disk space users get for storing their OVAs in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit</code></td>
<td>the maximum number of IOPS available under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td><code>virtual_servers_resource</code></td>
<td><code>limit</code></td>
<td>the maximum number of virtual servers users can create in the cloud. This parameter affects the number of virtual servers, VSs in Federation and load balancers users can create (VS/hour)</td>
</tr>
<tr>
<td><code>autoscaled_servers_resource</code></td>
<td><code>limit</code></td>
<td>the maximum number of VSs for which the user can enable autoscaling under this bucket (VS/hour)</td>
</tr>
<tr>
<td><code>templates_resource</code></td>
<td><code>limit</code></td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td><code>compute_resource_storing_resource</code></td>
<td><code>limit</code></td>
<td>the total amount of disk space users can request for storing their backups, ISOs, and templates under this bucket (GB/hour)</td>
</tr>
</tbody>
</table>

*the `limit_default_cpu_share` parameter.*
### Virtual Server Preferences

<table>
<thead>
<tr>
<th>Resource</th>
<th>Limit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backups_resource</td>
<td>limit</td>
<td>the maximum number of backups users can create under this bucket (backup/hour)</td>
</tr>
<tr>
<td>iso_templates_resource</td>
<td>limit</td>
<td>the maximum number of ISO templates users can create under this bucket. (ISO/hour)</td>
</tr>
<tr>
<td>application_servers_resource</td>
<td>limit</td>
<td>the maximum number of application servers in the cloud that the users can create under this bucket. (application VS/hour)</td>
</tr>
<tr>
<td>container_servers_resource</td>
<td>limit</td>
<td>the maximum number of container servers in the cloud that the users can create under this bucket. (container VS/hour)</td>
</tr>
<tr>
<td>data_store_zone_resource</td>
<td>limit</td>
<td>the maximum amount of disk space (GB) users can request in the data store zone under the bucket. (GB/hour)</td>
</tr>
</tbody>
</table>

#### Page History

- **v. 6.2 Edge 1**
  - removed the `apply_to_all_resources_in_the_bucket` parameter

- **v. 6.0**
  - removed the `accelerated_servers_resource` parameter

- **v. 5.9**
  - added the `apply_to_all_resources_in_the_bucket` parameter

- **v. 5.7**
  - added the `preferences` parameter
  - added the `preconfigured_servers_resource` resource type that can have the following parameters:
    - `hypervisor_group_ids`
    - `data_store_group_ids`
    - `network_group_ids`

#### 22.7.8 Add Access Control for Other Server Type

To create new access control, use the following request:

POST /billing/buckets/:bucket_id/access_controls.xml
POST /billing/buckets/:bucket_id/access_controls.json

**XML Request Example**
curl -i -X POST -u user:userpass --url http://onapp.test/billing/buckets/331/access_controls.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<access_control><bucket_id>331</bucket_id><server_type>other</server_type><target_id>2</target_id><create_rate_card>false</create_rate_card><type>recipe_groups_resource</type><limits></limits><preferences></preferences></access_control>'

JSON Request Example

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/billing/buckets/331/access_controls.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"access_control": {"bucket_id": 331, "server_type": "other", "target_id": 2, "create_rate_card": false, "type": "recipe_groups_resource", "limits": {}, "preferences": {}}}'}
```

Where:
- **bucket_id**: the ID of the bucket with which this access control is associated.
- **server_type**: the server type this access control is applicable to, in this case other.
- **target_id**: the ID of the resource added to the Access Control.
- **create_rate_card**: set true if you want to add this resource not only to the Access Control, but to the Rate Card as well, otherwise, set false.
- **type**: the type of the resource for which configuration is set, it can be one of the following values:
  - **backup_resource_zone_resource**
  - **template_resource**
  - **edge_groups_resource**
  - **cdn_bandwidth_resource**
  - **recipe_groups_resource**
  - **service_addon_resource**
  - **blueprint_groups_resource**
- **target_name**: the name of the resource that is added to the access control. For example, it can be a label of an edge group.
- **preferences**: this parameter is applicable only to access control for virtual server type.
- **limits**: the array of limits for the resource. The access control for Other resources includes the **cdn_bandwidth_resource** limit allocated in GB per month.

**Page History**

- v. 6.3 Edge 1
  - added the **blueprint_groups_resource** resource type
- v. 5.8
  - added the following resource types:
    - **backup_resource_zone_resource**
22.7.9 Edit Access Control for Baremetal Server Type

To edit an access control, use the following request:

PUT /billing/buckets/:bucket_id/access_controls.xml
PUT /billing/buckets/:bucket_id/access_controls.json

XML Request Example

```bash
curl "http://onapp.test/billing/buckets/34/access_controls.xml"  
<access_control><type>bare_metal_servers_resource</type><bucket_id>34</bucket_id><server_type>baremetal</server_type><target_id>null</target_id><limits><limit>24</limit></limits></access_control>' -X PUT \
-u user:userpass \-H "Accept: application/xml" \-H "Content-Type: application/xml"
```

JSON Request Example

```bash
curl "http://onapp.test/billing/buckets/34/access_controls.json" -d  
'{"access_control": {"type": "bare_metal_servers_resource", "bucket_id":  
34, "server_type": "baremetal", "target_id": null, "limits": {"limit":  
24}}}'} -X PUT \
-u user:userpass \-H "Accept: application/json" \-H "Content-Type: application/json"
```

Where:

- **type** - the type of the resource for which configuration is set, it can be one of the following values:
  - `compute_zone_resource`
  - `bare_metal_servers_resource`
  - `network_zone_resource`

- **bucket_id** - the ID of the bucket with which this access control is associated

- **server_type** - the server type this access control is applicable to. Can be `virtual`, `smart`, `baremetal` or `vpc`.

- **target_id** - the ID of the resource that is edited in the access control

- **limits** - the array of limits for compute zone resources. Depending on the type of resource, you will have the following parameters:
  - for the `bare_metal_servers_resource` resource: `limit` - the total amount of baremetal servers available under this bucket
  - for the `compute_zone_resource` resource: `target_name` - a label of a compute zone that you want to add to access control
  - for the `network_zone_resource` resource: `limit_ip` - the total amount of IP addresses available under this bucket

Page History

v. 6.2 Edge 1
- removed the `apply_to_all_resources_in_the_bucket` parameter
v. 5.9

- added the `apply_to_all_resources_in_the_bucket` parameter

### 22.7.10 Edit Access Control for Smart Server Type

To edit access control, use the following request:

PUT /billing/buckets/:bucket_id/access_controls.xml

PUT /billing/buckets/:bucket_id/access_controls.json

**XML Request Example**

```
curl -X PUT http://onapp.test/billing/buckets/5/access_controls.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml' -u user:userpass -d '<access-control><bucket_id type="integer">5</bucket_id><server_type>smart</server_type><target_id type="integer">6</target_id><backup_server_zone_resource type="type">hourly</backup_server_zone_resource><target_name>BackupSmartZone</target_name><limits><limit_backup type="decimal">10.0</limit_backup><limit_backup_disk_size type="decimal">150.0</limit_backup_disk_size><limit_template type="decimal">10.0</limit_template><limit_template_disk_size type="decimal">10.0</limit_template_disk_size></limits></access_control>'
```

**JSON Request Example**

```
curl -X PUT http://onapp.test/billing/buckets/5/access_controls.json -H 'Accept: application/json' -H 'Content-Type: application/json' -u user:userpass -d '{"bucket_id": 5, "server_type": "smart", "target_id": 6, "type": "backup_server_zone_resource", "timing_strategy": "hourly", "target_name": "BackupSmartZone", "limits": {"limit_backup": 10.0, "limit_backup_disk_size": 150.0, "limit_template": 10.0, "limit_template_disk_size": 10.0}}'
```

**Where:**

- **bucket_id** * - the ID of the bucket with which this access control is associated
- **server_type** * - the server type this access control is applicable to (smart for Smart server type)
- **target_id** - the ID of the resource that is added to the access control
- **type** * - the type of the resource added to the access control, it can be one of the following values:
  - backups_resource
  - backup_server_zone_resource
  - compute_resource_storing_resource
  - compute_zone_resource
  - data_store_zone_resource
  - network_zone_resource
  - smart_servers_resource
- **timing_strategy** - the type of billing for each resource (hourly for Smart server type)
target_name - the name of the resource that is added to the access control. For example, it can be a label of a network or compute zone, etc.

limits - the array of limits for the resource

<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backups_resource</td>
<td>limit</td>
<td>the maximum number of backups users can create under this bucket (backup/hour)</td>
</tr>
<tr>
<td>backup_server_zone_resource</td>
<td>limit_backup</td>
<td>the maximum amount of backups users can create in this backup server zone under the bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_backup_disk_size</td>
<td>the maximum amount of disk space users get for storing their backups in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_template</td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_template_disk_size</td>
<td>the maximum amount of disk space users get for storing their templates in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td>compute_resource_storing_resource</td>
<td>limit</td>
<td>the total amount of disk space users can request for storing their backups, ISOs and templates under this bucket (GB/hour)</td>
</tr>
<tr>
<td>compute_zone_resource</td>
<td>limit_cpu</td>
<td>the maximum number of CPU cores that can be set in the smart server creation wizard when the user adds a server under this bucket in the compute zone (CPU core/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_cpu_share</td>
<td>the maximum amount of CPU shares users can request for all their servers in this compute zone under this bucket (CPU share %/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_cpu_units</td>
<td>the maximum amount of CPU units that users can request for all their smart servers within this compute zone under the bucket (CPU unit/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_memory</td>
<td>the maximum amount of RAM that users can request for all</td>
</tr>
<tr>
<td>Smart Server Limits</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>use_cpu_units</strong></td>
<td>Set to &quot;1&quot; to use CPU units instead of CPU shares (CPU unit/hour), otherwise, set to &quot;0&quot;. Set the amount of CPU units available to users under this bucket using the limit_cpu_units parameter.</td>
<td></td>
</tr>
<tr>
<td><strong>data_store_zone_resource</strong></td>
<td>limit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the maximum amount of disk space (GB) users can request in the data store zone under the bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>network_zone_resource</strong></td>
<td>limit_ip</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the maximum number of IP addresses users can request under this bucket (IP/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>limit_rate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the maximum port speed user can request in this network zone under the bucket (Mbps/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>smart_servers_resource</strong></td>
<td>limit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the maximum number of smart servers users can create in the cloud</td>
<td></td>
</tr>
<tr>
<td></td>
<td>This parameter affects the number of smart servers, smart servers in Federation and load balancers users can create (VS/hour).</td>
<td></td>
</tr>
</tbody>
</table>

**Page History**

- **v.6.2 Edge 1**
  - removed the apply_to_all_resources_in_the_bucket parameter

- **v.5.9**
  - added the apply_to_all_resources_in_the_bucket parameter

### 22.7.11 Edit Access Control for Virtual Server Type

To edit access control, use the following request:

```plaintext
PUT /billing/buckets/:bucket_id/access_controls.xml
PUT /billing/buckets/:bucket_id/access_controls.json
```

**XML Request Example**
curl "http://onapp.test/billing/buckets/5/access_controls.xml" -d '
<access-control><bucket_id type="integer">5</bucket_id><server_type>virtual</server_type><target_id type="integer">6</target_id><backup_server_zone_resource><target_name>BackupServerZone</target_name><preferences></preferences><limits><limit_backup type="decimal">10.0</limit_backup><limit_backup_disk_size type="decimal">150.0</limit_backup_disk_size><limit_template_type="decimal">10.0</limit_template_type><limit_template_disk_size type="decimal">10.0</limit_template_disk_size></limits></access_control>'
-X PUT \ -u user:userpass \ -H "Accept: application/xml" \ -H "Content-Type: application/xml"

JSON Request Example

curl "http://onapp.test/billing/buckets/5/access_controls.json" -d '
{"bucket_id": 5, "server_type": "virtual", "target_id": 6, "type": "backup_server_zone_resource", "target_name": "BackupServerZone", "limits": {"limit_backup": 10.0, "limit_backup_disk_size": 150.0, "limit_template": 10.0, "limit_template_disk_size": 10.0}}' -X PUT \ -u user:userpass \ -H "Accept: application/json" \ -H "Content-Type: application/json"

Where:

* **bucket_id** - the ID of the bucket with which this access control is associated
* **server_type** - the server type this access control is applicable to (virtual for Virtual server type)
* **target_id** - the ID of the resource that is added to the access control
* **type** - the type of the resource that is added to the access control, it can be one of the following values:
  * network_zone_resource
  * compute_zone_resource
  * backup_server_zone_resource
  * solidfire_data_store_zone_resource
  * virtual_servers_resource
  * autoscaled_servers_resource
  * templates_resource
  * compute_resource_storing_resource
  * backups_resource
- iso_templates_resource
- application_servers_resource
- container_servers_resource
- data_store_zone_resource
- preconfigured_servers_resource

**timing_strategy** - the type of billing for each resource: *hourly* or *monthly* (on peak usage)

**target_name** - the name of the resource that is added to the access control. For example, it can be a label of a network or compute zone, etc.

**preferences** - the array of zone IDs added to an instance package. This parameter is available for the preconfigured_servers_resource resource type.

**limits** - the array of limits for the resource

<table>
<thead>
<tr>
<th>Virtual Server Preferences Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>preconfigured_servers_resource</td>
<td>hypervisor_group_ids</td>
<td>The ID(s) of a compute zone added to an instance package.</td>
</tr>
<tr>
<td></td>
<td>data_store_group_ids</td>
<td>The ID(s) of a data store zone added to an instance package.</td>
</tr>
<tr>
<td></td>
<td>network_group_ids</td>
<td>The ID(s) of a network zone added to an instance package.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Virtual Server Limits Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>network_zone_resource</td>
<td>limit_ip</td>
<td>the maximum number of IP addresses users can request under this bucket (IP/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_rate</td>
<td>the maximum port speed amount user can request in this network zone under the bucket (Mbps/hour)</td>
</tr>
</tbody>
</table>

| compute_zone_resource      | limit_cpu  | the maximum amount of CPU cores that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU core/hour) |
|                            | limit_cpu_share | the maximum amount of CPU shares users can request for all their servers in this compute zone under this bucket (CPU share %/hour) |
|                            | limit_cpu_units | the maximum amount of CPU units that users can request for all their VSS within this compute zone under the bucket (CPU unit/hour) |
|                            | limit_memory | the maximum amount of RAM that users can request for all their VSSs |


<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>limit_default_cpu</code></td>
<td>the default amount of CPU cores that will be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td><code>limit_min_cpu</code></td>
<td>the minimum amount of CPU cores that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU core/hour)</td>
</tr>
<tr>
<td><code>limit_min_memory</code></td>
<td>the minimum amount of RAM that can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (GB/hour)</td>
</tr>
<tr>
<td><code>limit_default_cpu_share</code></td>
<td>the default amount of CPU shares that will be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU share %/hour)</td>
</tr>
<tr>
<td><code>limit_min_cpu_priority</code></td>
<td>the minimum amount of CPU priority which can be set in the VS creation wizard when the user adds a VS under this bucket in the compute zone (CPU share %/hour)</td>
</tr>
<tr>
<td><code>use_cpu_units</code></td>
<td>set to &quot;1&quot; to use CPU shares instead of CPU priority (CPU unit/hour). Otherwise, set to &quot;0&quot;. Set the amount of CPU units available to users under this bucket using the <code>limit_cpu_units</code> parameter.</td>
</tr>
<tr>
<td><code>use_default_cpu</code></td>
<td>set to &quot;1&quot; if a default amount of CPU cores is to be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU core/hour). Otherwise, set to &quot;0&quot;. Set the default amount of CPU cores using the <code>limit_default_cpu</code> parameter.</td>
</tr>
<tr>
<td><code>use_default_cpu_share</code></td>
<td>set to &quot;1&quot; if a default amount of CPU shares is to be set in the VS creation wizard when the user adds a VS in this compute zone under this bucket (CPU share %/hour). Otherwise, set to &quot;0&quot;. Set the default amount of CPU shares using the <code>limit_default_cpu_share</code> parameter.</td>
</tr>
</tbody>
</table>
## Virtual Server Preferences

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Limit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backup_server_zone_resource</td>
<td>limit_backup</td>
<td>the maximum amount of backups users can create in this backup server zone under the bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_backup_disk_size</td>
<td>the maximum amount of disk space users get for storing their backups in this backup server zone under the bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_template</td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_template_disk_size</td>
<td>the maximum amount of disk space users get for storing their templates in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_ova</td>
<td>the maximum amount of OVAs users can create in this backup server zone under the bucket (OVA/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_ova_disk_size</td>
<td>the maximum amount of disk space users get for storing their OVAs in this backup server zone under the bucket (GB/hour)</td>
</tr>
<tr>
<td>solidfire_data_store_zone_resource</td>
<td>limit</td>
<td>the maximum number of IOPS available under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>virtual_servers_resource</td>
<td>limit</td>
<td>the maximum number of virtual servers users can create in the cloud. This parameter affects the number of virtual servers, VSS in Federation and load balancers users can create (VS/hour)</td>
</tr>
<tr>
<td>autoscaled_servers_resource</td>
<td>limit</td>
<td>the maximum number of VSs for which the user can enable autoscaling under this bucket (VS/hour)</td>
</tr>
<tr>
<td>templates_resource</td>
<td>limit</td>
<td>the maximum amount of templates users can create in this backup server zone under the bucket (template/hour)</td>
</tr>
<tr>
<td>compute_resource_storing_resource</td>
<td>limit</td>
<td>the total amount of disk space users can request for storing their backups, ISOs, and templates under this bucket (GB/hour)</td>
</tr>
<tr>
<td>backups_resource</td>
<td>limit</td>
<td>the maximum number of backups users can create under this bucket (backup/hour)</td>
</tr>
</tbody>
</table>
Virtual Server Preferences

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>iso_templates_resource</td>
<td>limit</td>
<td>the maximum number of ISO templates users can create under this bucket. (ISO/hour)</td>
</tr>
<tr>
<td>application_servers_resource</td>
<td>limit</td>
<td>the maximum number of application servers in the cloud that the users can create under this bucket. (application VS/hour)</td>
</tr>
<tr>
<td>container_servers_resource</td>
<td>limit</td>
<td>the maximum number of container servers in the cloud that the users can create under this bucket. (container VS/hour)</td>
</tr>
<tr>
<td>data_store_zone_resource</td>
<td>limit</td>
<td>the maximum amount of disk space (GB) users can request in the data store zone under the bucket. (GB/hour)</td>
</tr>
</tbody>
</table>

Page History
v. 6.2 Edge 1
- removed the apply_to_all_resources_in_the_bucket parameter

v. 6.0
- removed the accelerated_servers_resource parameter

v. 5.9
- added the apply_to_all_resources_in_the_bucket parameter

v. 5.7
- added the preferences parameter
- added the preconfigured_servers_resource resource type that can have the following parameters:
  - hypervisor_group_ids
  - data_store_group_ids
  - network_group_ids

22.7.12 Edit Access Control for Other Server Type
To edit access control, use the following request:

PUT /billing/buckets/:bucket_id/access_controls.xml
PUT /billing/buckets/:bucket_id/access_controls.json

XML Request Example
curl -i -X PUT http://onapp.test/billing/buckets/331/access_controls.xml -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml' -d '<access_control><bucket_id>331</bucket_id><server_type>other</server_type><target_id>2</target_id><create_rate_card>false</create_rate_card><type>edge_groups_resource</type><target_name>EdgeGroup1</target_name></access_control>'

JSON Request Example

curl -i -X PUT http://onapp.test/billing/buckets/331/access_controls.json -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json' -d '{"access_control": {"bucket_id": 331, "server_type": "other", "target_id": 2, "type": "edge_groups_resource", "target_name": "EdgeGroup1"}}'

Where:

**bucket_id** - the ID of the bucket with which this access control is associated

**server_type** - the server type this access control is applicable to, in this case, other

**target_id** - the ID of the resource for which configuration is set

**type** - the type of the resource for which configuration is set. It can be one of the following values:

- `backup_resource_zone_resource`
- `template_resource`
- `edge_groups_resource`
- `cdn_bandwidth_resource`
- `recipe_groups_resource`
- `service_addon_resource`
- `blueprint_groups_resource`

**target_name** - the name of the resource that is added to the access control. For example, it can be a label of an edge group.

**preferences** - this parameter is applicable only to access control for virtual server type.

**limits** - the array of limits for the resource. The access control for Other resources includes the `cdn_bandwidth_resource` limit allocated in GB per month.

**Page History**

- v. 6.3 Edge 1
  - added the `blueprint_groups_resource` resource type
- v. 5.8
  - added the following resource types:
22.7.13 Delete Resource from Access Control for Baremetal Server Type

If you remove a compute/data store/network/backup server zone from the Access Control, it will not be possible to edit the resources of the servers running in this zone(s).

To delete resources from access controls, use the following request:

DELETE /billing/buckets/:bucket_id/access_controls.xml
DELETE /billing/buckets/:bucket_id/access_controls.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**Where:**

- **type** - the type of the resource for which configuration is set, it can be one of the following values:
  - compute_zone_resource
  - bare_metal_servers_resource
  - network_zone_resource

- **bucket_id** - the ID of the bucket with which this access control is associated

- **server_type** - the server type this access control is applicable to. Can be virtual, smart, baremetal or vpc.

- **target_id** - the ID of the resource which is deleted.
22.7.14 Delete Resource from Access Control for Smart Server Type

If you remove a compute/data store/network/backup server zone from the Access Control, it will not be possible to edit the resources of the servers running in this zone(s).

To delete resources from access controls, use the following request:

DELETE /billing/buckets/:bucket_id/access_controls.xml
DELETE /billing/buckets/:bucket_id/access_controls.json

XML Request Example

```bash
curl -i -X DELETE
http://onapp.test/billing/buckets/5/access_controls/delete.xml
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
-u user:userpass
-d '<access_control><type>backup_server_zone_resource</type><bucket_id>5</bucket_id><server_type>smart</server_type><target_id>105</target_id></access_control>'
```

JSON Request Example

```bash
curl -i -X DELETE
http://onapp.test/billing/buckets/5/access_controls/delete.json
-H 'Accept: application/json'
-H 'Content-type: application/json'
-u user:userpass
-d '{"access_control": {"type": "backup_server_zone_resource", "bucket_id": 5, "server_type": "smart", "target_id": 105}}'
```

Where:

- **type** - the type of the resource to be deleted, it can be one of the following values:
  - backups_resource
  - backup_server_zone_resource
  - compute_resource_storing_resource
  - compute_zone_resource
  - data_store_zone_resource
  - network_zone_resource
  - smart_servers_resource

- **bucket_id** - the ID of the bucket with which this access control is associated

- **server_type** - the server type this access control is applicable to (smart for Smart server type)

- **target_id** - the ID of the resource that is deleted
22.7.15 Delete Resource from Access Control for Virtual Server Type

If you remove a compute/data store/network/backup server zone from the Access Control, it will not be possible to edit the resources of the servers running in this zone(s).

To delete resources from access controls, use the following request:

DELETE /billing/buckets/:bucket_id/access_controls.xml
DELETE /billing/buckets/:bucket_id/access_controls.json

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url  
http://onapp.test/billing/buckets/344/access_controls/delete.xml  -H  
'Accept: application/xml' -H 'Content-type: application/xml' -d  
'<access_control><type>backup_server_zone_resource</type><bucket_id>344</bucket_id><server_type>virtual</server_type><target_id>105</target_id></access_control>'
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass --url  
http://onapp.test/billing/buckets/344/access_controls/delete.json  -H  
'Accept: application/json' -H 'Content-type: application/json' -d  
'{"access_control": {"type": "backup_server_zone_resource", "bucket_id": 344, "server_type": "virtual", "target_id": 105}}'
```

Where:

type - the type of the resource to be deleted, it can be one of the following values:

- `network_zone_resource`
- `compute_zone_resource`
- `backup_server_zone_resource`
- `solidfire_data_store_zone_resource`
- `virtual_servers_resource`
- `autoscaled_servers_resource`
- `templates_resource`
- `compute_resource_storing_resource`
- backups_resource
- iso_templates_resource
- application_servers_resource
- container_servers_resource
- data_store_zone_resource
- preconfigured_servers_resource

`bucket_id` - the ID of the bucket with which this access control is associated
`server_type` - the server type this access control is applicable to (virtual for Virtual server type)
`target_id` - the ID of the resource that is deleted

**22.7.16 Delete Resource from Access Control for Other Server Type**

To delete resources from access controls, use the following request:

```
DELETE /billing/buckets/:bucket_id/access_controls.xml
DELETE /billing/buckets/:bucket_id/access_controls.json
```

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url
   http://onapp.test/billing/buckets/331/access_controls/delete.xml
   -H 'Accept: application/xml' -H 'Content-type: application/xml'
   -d '<access_control><type>recipe_groups_resource</type><bucket_id>331</bucket_id><server_type>other</server_type><target_id>2</target_id></access_control>
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass --url
   http://onapp.test/billing/buckets/331/access_controls/delete.json
   -H 'Accept: application/json' -H 'Content-type: application/json'
   -d '{"access_control": {"type": "recipe_groups_resource", "bucket_id": 331,
   "server_type": "other", "target_id": 2}}'
```

**Where:**

`type` - the type of the resource for which configuration is set, it can be one of the following values:

- backup_resource_zone_resource
- template_resource
- edge_groups_resource
- cdn_bandwidth_resource
• recipe_groups_resource

• service_addon_resource

**bucket_id** - the ID of the bucket with which this Access Control is associated

**server_type** - the server type this access control is applicable to, in this case, *other*

**target_id** - the ID of the resource which is deleted

Page History

v. 5.8

• added the following resource types:
  • backup_resource_zone_resource
  • cdn_bandwidth_resource

22.8 Rate Card

Rate Cards are the part of buckets that contain the free limits for resources and prices for resource usage. This section contains the API requests which you can use to manage Rate Cards.

22.8.1 Get List of Rate Cards for Baremetal Server Type

To get the list of rate cards, use the following request:

GET /billing/buckets/:bucket_id/rate_cards.xml
GET /billing/buckets/:bucket_id/rate_cards.json

**XML Request Example**

```shell
curl "http://onapp.test/billing/buckets/4/rate_cards.xml" -X GET \n-u user:userpass
```

**JSON Request Example**

```shell
curl "http://onapp.test/billing/buckets/4/rate_cards.json" -X GET \n-u user:userpass
```

**XML Output Example**
<rate_cards type="array">
    <rate_card>
        <bucket_id type="integer">4</bucket_id>
        <server_type>baremetal</server_type>
        <target_id type="integer">14</target_id>
        <type>network_zone_resource</type>
        <timing_strategy>hourly</timing_strategy>
        <target_name>Network baremetal zone</target_name>
        <prices>
            <limit_ip_free type="decimal">12</limit_ip_free>
            <price_ip type="decimal">23</price_ip>
        </prices>
    </rate_card>
    <rate_card>...</rate_card>
</rate_cards>

Where:

bucket_id - the ID of the bucket with which this rate card is associated.

server_type - the server type this rate card is applicable to. Can be virtual, smart, baremetal or vpc.

target_id - the ID of the resource for which the prices are set.

type - the type of the resource for which configuration is set. The value can be network_zone_resource.

timing_strategy - the type of billing for each resource: hourly or monthly (on peak usage).

target_name - the name of the resource that is added to the bucket. For example, this can be the label of a template group or a compute zone, etc.

prices - the price for network zone resources:

    limit_ip_free - the number of IP addresses users can request for free either per hour or per month

    price_ip - the price per IP address per hour charged for VSs which are built in this network zone under this bucket.

22.8.2 Get List of Rate Cards for Smart Server Type

To get the list of rate cards, use the following request:

GET /billing/buckets/:bucket_id/rate_cards.xml
GET /billing/buckets/:bucket_id/rate_cards.json

XML Request Example

curl -X GET http://onapp.test/billing/buckets/5/rate_cards.xml -u user:userpass

JSON Request Example
curl -X GET http://onapp.test/billing/buckets/5/rate_cards.json -u user:userpass

XML Output Example

```xml
<rate_cards type="array">
  <rate_card>
    <bucket_id type="integer">5</bucket_id>
    <server_type>smart</server_type>
    <target_id type="integer">7</target_id>
    <type>network_zone_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>Smart Network Zone</target_name>
    <prices>
      <price_data_sent type="decimal">10.0</price_data_sent>
      <price_data_received type="decimal">1.0</price_data_received>
    </prices>
  </rate_card>
  <rate_card>...
</rate_cards>
```

Where:

- **bucket_id** - the ID of the bucket with which this rate card is associated
- **server_type** - the server type this rate card is applicable to (smart for Smart server type)
- **target_id** - the ID of the resource that is added to the rate card
- **type** - the type of the resource that is added to the rate card, it can be one of the following values:
  - backups_resource
  - backup_server_zone_resource
  - compute_resource_storing_resource
  - compute_zone_resource
  - data_store_zone_resource
  - network_zone_resource
  - smart_servers_resource
- **timing_strategy** - the type of billing for each resource (hourly for Smart server type)
- **target_name** - the name of the resource that is added to the rate card. For example, it can be a label of a network or compute zone, etc.
- **prices** - the array of resource prices and limits

<table>
<thead>
<tr>
<th>Smart Server Type</th>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backups_resource</td>
<td></td>
<td>limit_free</td>
<td>the number of backups users can create for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>price</td>
<td>the price per backup created by the user under</td>
</tr>
<tr>
<td>Smart Server Type</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>backup_server_zone_resource</strong></td>
<td>this bucket per hour (backup/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>limit_backup_free</strong></td>
<td>the amount of backups users can store in this backup server zone for free under this bucket (backup/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>limit_backup_disk_size_free</strong></td>
<td>the amount of disk space users can request for free to store their backups in this backup server zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>limit_template_free</strong></td>
<td>the amount of templates users can store in this backup server zone for free under this bucket (template/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>limit_template_disk_size_free</strong></td>
<td>the amount of disk space users can request for free to store their templates in this backup server zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_backup</strong></td>
<td>the price per backup per hour, charged for the backups stored in this backup server zone under this bucket (backup/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_backup_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user’s backups stored in this backup server zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_template</strong></td>
<td>the price per template per hour, charged for the backups stored on this backup server zone under this bucket (template/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_template_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user’s templates stored in this backup server zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>compute_resource_storing_resource limit_free</strong></td>
<td>the amount of free disk space users can allocate to storing backups, ISOs and templates together (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smart Server Type</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>price</td>
<td>the price per GB per hour of disk space the user allocates to storing backups, ISOs, and templates (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>compute_zone_resource</td>
<td>limit_free_cpu</td>
<td>the amount of CPU cores users can request for free for the total number of smart servers built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>limit_free_cpu_share</td>
<td>the amount of CPU shares users can request for free for the total number of their smart servers built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>limit_free_cpu_units</td>
<td>the amount of CPU units users can request for free for the total number of their smart servers built in this compute zone under this bucket (unit/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>limit_free_memory</td>
<td>the amount of RAM users can request for free for the total number of their smart servers built in this compute zone under this bucket (Mb/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>price_on_cpu</td>
<td>the price per CPU core per hour, charged for powered on smart servers which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>price_off_cpu</td>
<td>the price per CPU core per hour, charged for powered off smart servers which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>price_on_cpu_share</td>
<td>the price for CPU shares, charged for powered on smart servers which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>price_off_cpu_share</td>
<td>the price for CPU shares, charged for powered off smart servers which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td>Smart Server Type</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>price_on_cpu_units</code></td>
<td>the price per CPU unit per hour, charged for powered on smart servers which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>price_off_cpu_units</code></td>
<td>the price per CPU unit per hour, charged for powered off smart servers which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>price_on_memory</code></td>
<td>the price for RAM, charged for powered on smart servers which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>price_off_memory</code></td>
<td>the price for RAM, charged for powered off smart servers which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>data_store_zone_resource</code></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>limit_free</code></td>
<td>the amount of disk space users can request for free per hour (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>limit_data_read_free</code></td>
<td>the amount of read data users can request for free per hour (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>limit_data_written_free</code></td>
<td>the amount of written data users can request for free per hour (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>limit_reads_completed_free</code></td>
<td>the amount of input requests users can request for free per hour (1M requests/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>limit_writes_completed_free</code></td>
<td>the amount of output requests users can request for free per hour (1M requests/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>price_on</code></td>
<td>the price per GB of disk space per hour, charged for powered on smart servers which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>price_off</code></td>
<td>the price per GB of disk space per hour, charged for powered off smart servers which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Smart Server Type</strong></td>
<td><strong>Definition</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_data_read</strong></td>
<td>the price per GB of read data per hour, charged for smart servers which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_data_written</strong></td>
<td>the price per GB of written data per hour, charged for smart servers which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_reads_completed</strong></td>
<td>the price per 1M input requests per hour, charged for smart servers which are built in this data store zone under this bucket (1M requests/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>price_writes_completed</strong></td>
<td>the price per 1M output requests per hour, charged for smart servers which are built in this data store zone under this bucket (1M requests/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>network_zone_resource</strong></td>
<td><strong>limit_rate_free</strong></td>
<td>the amount of port speed users can request for free for the total number of their smart servers built in this network zone under this bucket (Mbps/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>limit_ip_free</strong></td>
<td>the amount of IP addresses users can request for free per hour (IP/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>limit_data_sent_free</strong></td>
<td>the amount of data sent users can request for free per hour (GB/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>limit_data_received_free</strong></td>
<td>the amount of data received users can request for free per hour (GB/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>price_rate_on</strong></td>
<td>the price per Mbps of port speed per hour, charged for powered on smart servers which are built in this network zone under this bucket (Mbps/hour)</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>price_rate_off</strong></td>
<td>the price per Mbps of port speed per hour, charged</td>
<td></td>
</tr>
<tr>
<td>Smart Server Type</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>for powered off smart servers which are built in this network zone under this bucket (Mbps/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$price_{ip_on}$</td>
<td>the price per IP address per hour, charged for powered on smart servers which are built in this network zone under this bucket (IP/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$price_{ip_off}$</td>
<td>the price per IP address per hour, charged for powered off smart servers which are built in this network zone under this bucket (IP/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$price_{data_sent}$</td>
<td>the price per GB of data sent per hour, charged for smart servers which are built in this network zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$price_{data_received}$</td>
<td>the price per GB of data received per hour, charged for smart servers which are built in this network zone under this bucket (GB/hour)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 22.8.3 Get List of Rate Cards for Virtual Server Type

To get the list of rate cards, use the following request:

GET `/billing/buckets/:bucket_id/rate_cards.xml`
GET `/billing/buckets/:bucket_id/rate_cards.json`

XML Request Example

```bash
curl "http://onapp.test/billing/buckets/24/rate_cards.xml" -X GET \ -u user:userpass
```

JSON Request Example

```bash
curl "http://onapp.test/billing/buckets/24/rate_cards.json" -X GET \ -u user:userpass
```

XML Output Example
<rate_cards type="array">
  <rate_card>
    <bucket_id type="integer">24</bucket_id>
    <server_type>virtual</server_type>
    <target_id type="integer">12</target_id>
    <type>data_store_zone_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>Default DataStore Zone</target_name>
    <prices>
      <limit_free type="decimal">10.0</limit_free>
      <limit_data_read_free type="decimal">10.0</limit_data_read_free>
      <limit_data_written_free type="decimal">10.0</limit_data_written_free>
      <limit_reads_completed_free type="decimal">10.0</limit_reads_completed_free>
      <limit_writes_completed_free type="decimal">10.0</limit_writes_completed_free>
      <limit_free_monthly type="decimal">10.0</limit_free_monthly>
      <limit_data_read_free_monthly type="decimal">10.0</limit_data_read_free_monthly>
      <limit_data_written_free_monthly type="decimal">10.0</limit_data_written_free_monthly>
      <limit_reads_completed_free_monthly type="decimal">10.0</limit_reads_completed_free_monthly>
      <limit_writes_completed_free_monthly type="decimal">10.0</limit_writes_completed_free_monthly>
      <price_on type="decimal">10.0</price_on>
      <price_off type="decimal">10.0</price_off>
      <price_data_read type="decimal">10.0</price_data_read>
      <price_data_written type="decimal">10.0</price_data_written>
      <price_reads_completed type="decimal">10.0</price_reads_completed>
      <price_writes_completed type="decimal">10.0</price_writes_completed>
    </prices>
  </rate_card>
  <rate_card>...</rate_card>
</rate_cards>

Where:

bucket_id - the ID of the bucket with which this rate card is associated

server_type - the server type this rate card is applicable to (virtual for Virtual server type)

target_id - the ID of the resource that is added to the rate card

type - the type of the resource that is added to the rate card, it can be one of the following values:

- network_zone_resource
- compute_zone_resource
- backup_server_zone_resource
- solidfire_data_store_zone_resource
- autoscaled_servers_resource
- templates_resource
- `compute_resource_storing_resource`
- `backups_resource`
- `iso_templates_resource`
- `accelerated_servers_resource`
- `data_store_zone_resource`
- `preconfigured_servers_resource`

**timing_strategy** - the type of billing for each resource: *hourly* or *monthly* (on peak usage)

**target_name** - the name of the resource that is added to the rate card. For example, it can be a label of a network or compute zone, etc.

**prices** - the array of resource prices and limits

<table>
<thead>
<tr>
<th>Virtual Server</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>compute_zone_resource</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>limit_free_cpu</code></td>
<td>the amount of CPU cores users can request for free for the total number of VSs built in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_cpu_share</code></td>
<td>the amount of CPU shares users can request for free for the total number of their VSs built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_cpu_units</code></td>
<td>the amount of CPU units users can request for free for the total number of their VSs built in this compute zone under this bucket (unit/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_memory</code></td>
<td>the amount of RAM (GB/hr) users can request for free for the total number of their VSs built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price_on_cpu</code></td>
<td>the price per CPU core per hour,</td>
</tr>
<tr>
<td><strong>Virtual Server</strong></td>
<td>charged for powered on VSs which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>price_off_cpu</strong></td>
<td>the price per CPU core per hour, charged for powered off VSs which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_on_cpu_share</strong></td>
<td>the price for CPU shares, charged for powered on VSs which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_off_cpu_share</strong></td>
<td>the price for CPU shares, charged for powered off VSs which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_on_cpu_units</strong></td>
<td>the price per CPU unit per hour, charged for powered on VSs which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_off_cpu_units</strong></td>
<td>the price per CPU unit per hour, charged for powered off VSs which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_on_memory</strong></td>
<td>the price for RAM, charged for powered on VSs which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_off_memory</strong></td>
<td>the price for RAM, charged for powered off VSs which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>Data Store Zone Resource</td>
<td>Limit Description</td>
<td></td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------</td>
<td></td>
</tr>
<tr>
<td><code>data_store_zone_resource</code></td>
<td><strong>limit_free</strong> the amount of disk space users can request for free either per hour or per month (GB/limit_type)</td>
<td></td>
</tr>
<tr>
<td><code>limit_data_read_free</code></td>
<td>the amount of read data users can request for free either per hour or per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><code>limit_data_written_free</code></td>
<td>the amount of written data users can request for free either per hour or per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><code>limit_reads_completed_free</code></td>
<td>the amount of input requests users can request for free either per hour or per month (1M requests/hour)</td>
<td></td>
</tr>
<tr>
<td><code>limit_writes_completed_free</code></td>
<td>the price per 1M output requests per hour, charged for VSs which are built in this data store zone under this bucket (1M requests/hour)</td>
<td></td>
</tr>
<tr>
<td><code>limit_free_monthly</code></td>
<td>the amount of disk space users can request for free per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><code>limit_data_read_free_monthly</code></td>
<td>the amount of read data users can request for free per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><code>limit_data_written_free_monthly</code></td>
<td>the amount of written data users can request for free per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><code>limit_reads_completed_free_monthly</code></td>
<td>the amount of input requests users can request for free per month (1M requests/hour)</td>
<td></td>
</tr>
</tbody>
</table>
## Virtual Server

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>limit_writes_completed_free_monthly</td>
<td>the amount of output requests users can request for free either per hour or per month (1M requests/hour)</td>
</tr>
<tr>
<td>price_on</td>
<td>the price per GB of disk space per hour, charged for powered on VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_off</td>
<td>the price per GB of disk space per hour, charged for powered off VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_read</td>
<td>the price per GB of read data per hour, charged for VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_written</td>
<td>the price per GB of written data per hour, charged for VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_reads_completed</td>
<td>the price per 1M input requests per hour, charged for VSs which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>price_writes_completed</td>
<td>the price per 1M output requests per hour, charged for VSs which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>network_zone_resource</td>
<td>limit_rate_free</td>
</tr>
<tr>
<td></td>
<td>the amount of port speed users can request for free for</td>
</tr>
<tr>
<td><strong>Virtual Server</strong></td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>-</td>
<td>the total number of their VSs built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><code>limit_ip_free</code></td>
<td>the amount of IP address users can request for free either per hour or per month (IP/hour)</td>
</tr>
<tr>
<td><code>limit_data_sent_free</code></td>
<td>the amount of data sent users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td><code>limit_data_received_free</code></td>
<td>the amount of data received users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td><code>limit_ip_free_monthly</code></td>
<td>the amount of IP address users can request for free per month (IP/hour)</td>
</tr>
<tr>
<td><code>limit_data_sent_free_monthly</code></td>
<td>the amount of data sent users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td><code>limit_data_received_free_monthly</code></td>
<td>the amount of data received users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td><code>price_rate_on</code></td>
<td>the price per Mbps of port speed per hour, charged for powered on VSs which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><code>price_rate_off</code></td>
<td>the price per Mbps of port speed per hour, charged for powered off VSs which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><code>price_ip_on</code></td>
<td>the price per IP address per hour, charged for powered on VSs which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td><strong>Virtual Server</strong></td>
<td><strong>Virtual Server zone under this bucket (IP/hour)</strong></td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td><strong>price_ip_off</strong></td>
<td>the price per IP address per hour, charged for powered off VSs which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td><strong>price_data_sent</strong></td>
<td>the price per GB of data sent per hour, charged for VSs which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_data_received</strong></td>
<td>the price per GB of data received per hour, charged for VSs which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>backup_server_zone_resource</strong></td>
<td><strong>limit_backup_free</strong></td>
</tr>
<tr>
<td></td>
<td>the amount of backups users can store in this backup server zone for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td><strong>limit_backup_disk_size_free</strong></td>
</tr>
<tr>
<td></td>
<td>the amount of disk space users can request for free to store their backups in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><strong>limit_template_free</strong></td>
</tr>
<tr>
<td></td>
<td>the amount of templates users can store in this backup server zone for free under this bucket (template/hour)</td>
</tr>
<tr>
<td></td>
<td><strong>limit_template_disk_size_free</strong></td>
</tr>
<tr>
<td></td>
<td>the amount of disk space users can request for free to store their templates in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><strong>limit_ova_free</strong></td>
</tr>
<tr>
<td></td>
<td>the amount of OVA users can store in this backup server zone for free under</td>
</tr>
<tr>
<td><strong>Virtual Server</strong></td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>limit_ova_disk_size_free</strong></td>
<td>the amount of disk space users can request for free to store their OVAs in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_backup</strong></td>
<td>the price per backup per hour, charged for the backups stored in this backup server zone under this bucket (backup/hour)</td>
</tr>
<tr>
<td><strong>price_backup_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user's backups stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_template</strong></td>
<td>the price per template per hour, charged for the backups stored on this backup server zone under this bucket (template/hour)</td>
</tr>
<tr>
<td><strong>price_template_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user's templates stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_ova</strong></td>
<td>the price per OVA per hour, charged for the backups stored on this backup server zone under this bucket (OVA/hour)</td>
</tr>
<tr>
<td><strong>price_ova_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user's OVAs stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>draas_resource</strong></td>
<td>price_disk_size</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>price_memory</td>
<td>the additional price for RAM that applies to a virtual server with enabled DRaaS (GB/hour)</td>
</tr>
<tr>
<td>price_cpus</td>
<td>the additional price for CPU that applies to a virtual server with enabled DRaaS (core/hour)</td>
</tr>
<tr>
<td>price_cpu_shares</td>
<td>the additional price for CPU shares that applies to a virtual server with enabled DRaaS (%/hour)</td>
</tr>
<tr>
<td>price_cpu_units</td>
<td>the additional price for CPU units that applies to a virtual server with enabled DRaaS (unit/hour)</td>
</tr>
<tr>
<td>price_nodes</td>
<td>the additional price for nodes that applies to a virtual server with enabled DRaaS (node/hour)</td>
</tr>
<tr>
<td>compute_resource_storing_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td>price</td>
<td>the price per GB per hour of disk space the user allocates to storing backups, ISOs and templates (GB/hour)</td>
</tr>
<tr>
<td>backups_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td>price</td>
<td>the price per backup created by the user under this bucket per hour (backup/hour)</td>
</tr>
</tbody>
</table>
### Virtual Server

<table>
<thead>
<tr>
<th>resource</th>
<th>limit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>templates_resource</td>
<td>limit_free</td>
<td>the number of templates a user under this bucket can create for free (template/hour)</td>
</tr>
<tr>
<td></td>
<td>price</td>
<td>the price per template created by the user under this bucket per hour (template/hour)</td>
</tr>
<tr>
<td>iso_templates_resource</td>
<td>limit_free</td>
<td>the number of ISOs a user under this bucket can create for free (ISO/hour)</td>
</tr>
<tr>
<td></td>
<td>price</td>
<td>the price per ISO created by the user under this bucket per hour (ISO/hour)</td>
</tr>
<tr>
<td>accelerated_servers_resource</td>
<td>limit_free</td>
<td>the amount of virtual servers for which the user can enable acceleration for free under this bucket (VS/hour)</td>
</tr>
<tr>
<td></td>
<td>price</td>
<td>the price per VS per hour for VSs for which the user enables acceleration (VS/hour)</td>
</tr>
<tr>
<td>autoscaled_servers_resource</td>
<td>limit_free</td>
<td>the amount of virtual servers for which the user can enable autoscaling for free under this bucket (VS/hour)</td>
</tr>
<tr>
<td></td>
<td>price</td>
<td>the price per VS per hour for VSs for which the user enables autoscaling (VS/hour)</td>
</tr>
<tr>
<td>solidfire_data_store_zone_resource</td>
<td>limit_free</td>
<td>the number of IOPS requests a user under this bucket can create for free (IOPS/hour)</td>
</tr>
<tr>
<td></td>
<td>price_on</td>
<td>the price per GB of disk space per hour, charged for powered on VSs which are built in this data store</td>
</tr>
</tbody>
</table>
### Virtual Server

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zone under this bucket</td>
<td>(GB/hour)</td>
</tr>
<tr>
<td>price_off</td>
<td>the price per GB of disk space per hour, charged for powered off VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>preconfigured_servers_resource</td>
<td>price_on</td>
</tr>
<tr>
<td>price_off</td>
<td>the price per instance package per hour, charged for powered off VSs which are built on this instance package under this bucket</td>
</tr>
<tr>
<td>price_overused_bandwidth</td>
<td>the price per overused bandwidth per hour (GB/hr), charged for all VSs which are built on this instance package under this bucket</td>
</tr>
</tbody>
</table>

**Page History**

v. 5.7

- removed the legacy_resource_id parameter
- added the preconfigured_servers_resource parameter that can have the following parameters:
  - price_on
  - price_off
  - price_overused_bandwidth

22.8.4 Get List of Rate Cards for Other Server Type

To get the list of rate cards, use the following request:
GET /billing/buckets/:bucket_id/rate_cards.xml
GET /billing/buckets/:bucket_id/rate_cards.json

XML Request Example

curl "http://onapp.test/billing/buckets/5263/rate_cards.xml" -X GET \
-u user:userpass

JSON Request Example

curl "http://onapp.test/billing/buckets/5263/rate_cards.json" -X GET \
-u user:userpass

XML Output Example

```xml
<rate_cards type="array">
  <rate_card>
    <bucket_id type="integer">5263</bucket_id>
    <server_type>other</server_type>
    <target_id type="integer">2</target_id>
    <type>edge_groups_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>qaOHegF</target_name>
    <prices>
      <price type="decimal">0.0</price>
    </prices>
  </rate_card>
</rate_cards>
```

Where:

- **bucket_id** - the ID of the bucket with which this rate card is associated.
- **server_type** - the server type this rate card is applicable to. In this case *other*.
- **target_id** - the ID of the resource for which the prices are set.
- **type** - the type of the resource for which configuration is set. The value can be one of the following:
  - *backup_resource_zone_resource*
  - *edge_groups_resource*
  - *template_resource*
  - *service_addon_resource*

- **timing_strategy** - the type of billing for each resource: hourly or monthly.
- **target_name** - the name of the resource that was added to the bucket. For example, this can be the label of a template.
- **prices** - the array of prices and free limits for the resource that can be the following.

<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>backup_resource_zone_resource</em></td>
<td>price</td>
<td>- The price for a recovery point (backup) per hour.</td>
</tr>
<tr>
<td></td>
<td>price_recovery_point_size</td>
<td></td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>price_space_used limit_free</td>
<td>• The price for a recovery point (backup) size in Gb per hour.</td>
<td></td>
</tr>
<tr>
<td>limit_recovery_point_size_free</td>
<td>• The price for a total disk size (Gb/hour) taken by all backups on a particular virtual server.</td>
<td></td>
</tr>
<tr>
<td>limit_space_used_free</td>
<td>• The number of recovery points (backup/hour) users can store in a backup resource zone for free.</td>
<td></td>
</tr>
<tr>
<td>price_disk_size</td>
<td>• The size of backups (Gb/hour) on a particular virtual server that users can consume in a backup resource zone for free.</td>
<td></td>
</tr>
<tr>
<td>edge_groups_resource</td>
<td>price</td>
<td>The price per GB of bandwidth.</td>
</tr>
<tr>
<td>service_addon_resource</td>
<td>price</td>
<td>The price per service add-on per hour.</td>
</tr>
<tr>
<td>price_cpu</td>
<td>The price for the CPU usage per hour per CPU core.</td>
<td></td>
</tr>
<tr>
<td>price_memory</td>
<td>The additional price for RAM per GB per hour.</td>
<td></td>
</tr>
<tr>
<td>price_disk_size</td>
<td>The additional price for disk size per GB per hour.</td>
<td></td>
</tr>
<tr>
<td>template_resource</td>
<td>price</td>
<td>The price per template in a template store.</td>
</tr>
</tbody>
</table>

**Page History**

v. 6.1 Edge 2
- added the following parameters for a backup_resource_zone_resource type:
  - price_space_used
  - limit_space_used_free

v. 6.0
- added the following parameters for a backup_resource_zone_resource type:
  - price_recovery_point_size
  - limit_recovery_point_size_free

v. 5.8
- added the backup_resource_zone_resource type
22.8.5 Add Rate Cards for Baremetal Server Type

To add rate cards, use the following request:

POST /billing/buckets/:bucket_id/rate_cards.xml
POST /billing/buckets/:bucket_id/rate_cards.json

**XML Request Example**

```bash
curl "http://onapp.test/billing/buckets/35/rate_cards.xml" 
  -d '<?xml version="1.0" encoding="UTF-8"?>
      <rate_card>
        <target_id>67</target_id>
        <type>network_zone_resource</type>
        <bucket_id>35</bucket_id>
        <server_type>baremetal</server_type>
        <prices>
          <limit_ip_free>12</limit_ip_free>
          <price_ip>23</price_ip>
        </prices>
      </rate_card>'
  -X POST \
  -u user:userpass \
  -H "Accept: application/xml" \
  -H "Content-Type: application/xml"
```

**JSON Request Example**

```bash
curl "http://onapp.test/billing/buckets/35/rate_cards.json" -d 
  '{"rate_card": {
    "target_id": "67",
    "type": "network_zone_resource",
    "bucket_id": "35",
    "server_type": "baremetal",
    "prices": {
      "limit_ip_free": "12",
      "price_ip": "23"
    }
  }}' 
  -X POST \
  -u user:userpass \
  -H "Accept: application/json" \
  -H "Content-Type: application/json"
```

**Where:**

- `bucket_id` - the ID of the bucket with which this rate card is associated.
- `server_type` - the server type this rate card is applicable to. Can be virtual, smart, baremetal or vpc.
- `target_id` - the ID of the resource for which the prices are set.
- `type` - the type of the resource for which configuration is set. The value can be `network_zone_resource`.
- `prices` - the price for network zone resources:
  - `limit_ip_free` - the number of IP addresses users can request for free either per hour or per month
  - `price_ip` - the price per IP address per hour charged for VSs which are built in this network zone under this bucket

**Page History**

- v. 6.2 Edge 1
  - removed the `apply_to_all_resources_in_the_bucket` parameter
v. 5.9

- added the **apply_to_all_resources_in_the_bucket** parameter

### 22.8.6 Add Rate Cards for Smart Server Type

To add rate cards, use the following request:

**POST /billing/buckets/:bucket_id/rate_cards.xml**

**POST /billing/buckets/:bucket_id/rate_cards.json**

#### XML Request Example

```
curl -X POST http://onapp.test/billing/buckets/5/rate_cards.xml
-H 'Accept: application/xml' -H 'Content-Type: application/xml' -u
user:userpass -d '<rate_card><bucket_id type="integer">5</bucket_id><server_type>smart</server_type><target_id type="integer">7</target_id><type>network_zone_resource</type><timing_strategy>hourly</timing_strategy><target_name>Smart Network Zone</target_name><prices><price_data_sent type="decimal">10.0</price_data_sent><price_data_received type="decimal">1.0</price_data_received></prices></rate_card>'
```

#### JSON Request Example

```
curl -X POST http://onapp.test/billing/buckets/5/rate_cards.json
-H 'Accept: application/json' -H 'Content-Type: application/json' -u
user:userpass -d '{"bucket_id": 5, "server_type": "smart", "target_id": 7, "type": "network_zone_resource", "timing_strategy": "hourly", "target_name": "Smart Network Zone", "prices": {"price_data_sent": 10.0, "price_data_received": 1.0}}'
```

**Where:**

- **bucket_id** * - the ID of the bucket with which this rate card is associated
- **server_type** * - the server type this rate card is applicable to (smart for Smart server type)
- **target_id** - the ID of the resource that is added to the rate card
- **type** * - the type of the resource that is added to the rate card, it can be one of the following values:
  - backups_resource
  - backup_server_zone_resource
  - compute_resource_storing_resource
  - compute_zone_resource
  - data_store_zone_resource
  - network_zone_resource
  - smart_servers_resource
- **timing_strategy** - the type of billing for each resource (hourly for Smart server type)
- **target_name** - the name of the resource that is added to the rate card. For example, it can be a label of a network or compute zone, etc.
- **prices** - the array of resource prices and free limits
<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>backups_resource</code></td>
<td><code>limit_free</code></td>
<td>the number of backups users can create for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price</code></td>
<td>the price per backup created by the user under this bucket per hour (backup/hour)</td>
</tr>
<tr>
<td><code>backup_server_zone_resource</code></td>
<td><code>limit_backup_free</code></td>
<td>the amount of backups users can store in this backup server zone for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_backup_disk_size_free</code></td>
<td>the amount of disk space users can request for free to store their backups in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_template_free</code></td>
<td>the amount of templates users can store in this backup server zone for free under this bucket (template/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_template_disk_size_free</code></td>
<td>the amount of disk space users can request for free to store their templates in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price_backup</code></td>
<td>the price per backup per hour, charged for the backups stored in this backup server zone under this bucket (backup/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price_backup_disk_size</code></td>
<td>the price per GB per hour, charged for the disk size occupied by the user’s backups stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price_template</code></td>
<td>the price per template per hour, charged for the backups stored on this backup server zone under this bucket (template/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price_template_disk_size</code></td>
<td>the price per GB per hour, charged for the disk size</td>
</tr>
<tr>
<td>Smart Server Type</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>compute_resource_storing_resource limit_free</td>
<td>occupied by the user’s templates stored in this backup server zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>price</td>
<td>the price per GB per hour of disk space the user allocates to storing backups, ISOs, and templates together (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>compute_zone_resource limit_free_cpu</td>
<td>the amount of CPU cores users can request for free for the total number of smart servers built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_free_cpu_share</td>
<td>the amount of CPU shares users can request for free for the total number of their smart servers built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_free_cpu_units</td>
<td>the amount of CPU units users can request for free for the total number of their smart servers built in this compute zone under this bucket (unit/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_free_memory</td>
<td>the amount of RAM users can request for free for the total number of their smart servers built in this compute zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>price_on_cpu</td>
<td>the price per CPU core per hour, charged for powered on smart servers which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off_cpu</td>
<td>the price per CPU core per hour, charged for powered off smart servers which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
</tbody>
</table>
## Smart Server Type

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>price_on_cpu_share</td>
<td>the price for CPU shares, charged for powered on smart servers which are built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td>price_off_cpu_share</td>
<td>the price for CPU shares, charged for powered off smart servers which are built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td>price_on_cpu_units</td>
<td>the price per CPU unit per hour, charged for powered on smart servers which are built in this compute zone under this bucket (CPU unit/hour)</td>
</tr>
<tr>
<td>price_off_cpu_units</td>
<td>the price per CPU unit per hour, charged for powered off smart servers which are built in this compute zone under this bucket (CPU unit/hour)</td>
</tr>
<tr>
<td>price_on_memory</td>
<td>the price for RAM, charged for powered on smart servers which are built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_off_memory</td>
<td>the price for RAM, charged for powered off smart servers which are built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>data_store_zone_resource</td>
<td></td>
</tr>
<tr>
<td>limit_free</td>
<td>the amount of disk space users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td>limit_data_read_free</td>
<td>the amount of read data users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td>limit_data_written_free</td>
<td>the amount of written data users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td>limit_reads_completed_free</td>
<td>the amount of input requests users can request for free per hour (1M requests/hour)</td>
</tr>
<tr>
<td>limit_writes_completed_free</td>
<td>the amount of output requests users can request for free per hour (1M requests/hour)</td>
</tr>
<tr>
<td>Smart Server Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>price_on</td>
<td>the price per GB of disk space per hour, charged for powered on smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_off</td>
<td>the price per GB of disk space per hour, charged for powered off smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_read</td>
<td>the price per GB of read data per hour, charged for smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_written</td>
<td>the price per GB of written data per hour, charged for smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_reads_completed</td>
<td>the price per 1M input requests per hour, charged for smart servers which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>price_writes_completed</td>
<td>the price per 1M output requests per hour, charged for smart servers which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>network_zone_resource</td>
<td>limit_rate_free</td>
</tr>
<tr>
<td></td>
<td>the amount of port speed users can request for free for the total number of their smart servers built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_ip_free</td>
</tr>
<tr>
<td></td>
<td>the amount of IP addresses users can request for free per hour (IP/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_data_sent_free</td>
</tr>
<tr>
<td></td>
<td>the amount of data sent users can request for free per hour (GB/hour)</td>
</tr>
</tbody>
</table>
### Smart Server Type

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>limit_data_received_free</code></td>
<td>the amount of data received users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td><code>price_rate_on</code></td>
<td>the price per Mbps of port speed per hour, charged for powered on smart servers which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><code>price_rate_off</code></td>
<td>the price per Mbps of port speed per hour, charged for powered off smart servers which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><code>price_ip_on</code></td>
<td>the price per IP address per hour, charged for powered on smart servers which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td><code>price_ip_off</code></td>
<td>the price per IP address per hour, charged for powered off smart servers which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td><code>price_data_sent</code></td>
<td>the price per GB of data sent per hour, charged for smart servers which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><code>price_data_received</code></td>
<td>the price per GB of data received per hour, charged for smart servers which are built in this network zone under this bucket (GB/hour)</td>
</tr>
</tbody>
</table>

### Page History

- **v. 6.2 Edge 1**
  - removed the `apply_to_all_resources_in_the_bucket` parameter

- **v. 5.9**
  - added the `apply_to_all_resources_in_the_bucket` parameter

### 22.8.7 Add Rate Cards for Virtual Server Type

To add rate cards, use the following request:
POST /billing/buckets/:bucket_id/rate_cards.xml
POST /billing/buckets/:bucket_id/rate_cards.json

**XML Request Example**

```bash
curl "http://onapp.test/billing/buckets/2/rate_cards.xml" -d
    
'<rate_card>
    <bucket_id type="integer">2</bucket_id>
    <server_type>virtual</server_type>
    <target_id type="integer">12</target_id>
    <type>data_store_zone_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>Default DataStore Zone</target_name>
    <prices>
    <limit_free type="decimal">10.0</limit_free>
    <price_on type="decimal">2.0</price_on>
    <price_off type="decimal">1.0</price_off>
    </prices>
</rate_card>'

-X POST
-u user:userpass
-H "Accept: application/xml"
-H "Content-Type: application/xml"
```

**JSON Request Example**

```bash
curl "http://onapp.test/billing/buckets/2/rate_cards.json" -d
    
'{"bucket_id": 2,"server_type": "virtual","target_id": 12,"type": "data_store_zone_resource","timing_strategy": "hourly","prices": {"limit_free": 10,"price_on": 2,"price_off": 1}}' -X

POST
-u user:userpass
-H "Accept: application/json"
-H "Content-Type: application/json"
```

**Where:**

- **bucket_id** * - the ID of the bucket with which this rate card is associated
- **server_type** * - the server type this rate card is applicable to (virtual for Virtual server type)
- **target_id** - the ID of the resource that is added to the rate card
- **type** * - the type of the resource that is added to the rate card, it can be one of the following values:
  - `network_zone_resource`
  - `compute_zone_resource`
  - `backup_server_zone_resource`
  - `solidfire_data_store_zone_resource`
  - `autoscaled_servers_resource`
  - `templates_resource`
  - `compute_resource_storing_resource`
- `backups_resource`
- `iso_templates_resource`
- `accelerated_servers_resource`
- `data_store_zone_resource`
- `preconfigured_servers_resource`

`timing_strategy` - the type of billing for each resource: `hourly` or `monthly` (on peak usage)

`target_name` - the name of the resource that is added to the rate card. For example, it can be a label of a network or compute zone, etc.

`prices` - the array of resource prices and free limits

<table>
<thead>
<tr>
<th>Virtual Server Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>compute_zone_resource</code></td>
<td><code>limit_free_cpu</code></td>
<td>the amount of CPU cores users can request for free for the total number of VSs built in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_cpu_share</code></td>
<td>the amount of CPU shares users can request for free for the total number of their VSs built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_cpu_units</code></td>
<td>the amount of CPU units users can request for free for the total number of their VSs built in this compute zone under this bucket (unit/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_memory</code></td>
<td>the amount of RAM users can request for free for the total number of their VSs built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price_on_cpu</code></td>
<td>the price per CPU core per hour, charged for powered on VSs which are</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>price_off_cpu</td>
<td>the price per CPU core per hour, charged for powered off VSs which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td>price_on_cpu_share</td>
<td>the price for CPU shares, charged for powered on VSs which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off_cpu_share</td>
<td>the price for CPU shares, charged for powered off VSs which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td>price_on_cpu_units</td>
<td>the price per CPU unit per hour, charged for powered on VSs which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off_cpu_units</td>
<td>the price per CPU unit per hour, charged for powered off VSs which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
</tr>
<tr>
<td>price_on_memory</td>
<td>the price for RAM, charged for powered on VSs which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off_memory</td>
<td>the price for RAM, charged for powered off VSs which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
</tr>
</tbody>
</table>
### Virtual Server

<table>
<thead>
<tr>
<th>data_store_zone_resource</th>
<th>limit_free</th>
<th>the amount of disk space users can request for free either per hour or per month (GB/limit_type)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>limit_data_read_free</td>
<td>the amount of read data users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_data_written_free</td>
<td>the amount of written data users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_reads_completed_free</td>
<td>the amount of input requests users can request for free either per hour or per month (1M requests/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_writes_completed_free</td>
<td>the amount of output requests per hour users can request for free (1M requests/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_free_monthly</td>
<td>the amount of disk space users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_data_read_free_monthly</td>
<td>the amount of read data users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_data_written_free_monthly</td>
<td>the amount of written data users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_reads_completed_free_monthly</td>
<td>the amount of input requests users can request for free per month (1M requests/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_writes_completed_free_monthly</td>
<td>the amount of output requests users can request for free either per hour or per month (1M requests/hour)</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td><strong>price_on</strong></td>
<td>the price per GB of disk space per hour, charged for powered on VSs which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_off</strong></td>
<td>the price per GB of disk space per hour, charged for powered off VSs which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_data_read</strong></td>
<td>the price per GB of read data per hour, charged for VSs which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_data_written</strong></td>
<td>the price per GB of written data per hour, charged for VSs which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_reads_completed</strong></td>
<td>the price per 1M input requests per hour, charged for VSs which are built in this data store zone under this bucket (1M requests/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_writes_completed</strong></td>
<td>the price per 1M output requests per hour, charged for VSs which are built in this data store zone under this bucket (1M requests/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>network_zone_resource</strong></td>
<td>limit_rate_free</td>
<td></td>
</tr>
<tr>
<td><strong>limit_rate_free</strong></td>
<td>the amount of port speed users can request for free for the total number of their VSs built in this network zone under this bucket (Mbps/hour)</td>
<td></td>
</tr>
</tbody>
</table>
### Virtual Server

<table>
<thead>
<tr>
<th><strong>Variable</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>limit_ip_free</code></td>
<td>the amount of IP address users can request for free either per hour or per month (IP/hour)</td>
</tr>
<tr>
<td><code>limit_data_sent_free</code></td>
<td>the amount of data sent users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td><code>limit_data_received_free</code></td>
<td>the amount of data received users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td><code>limit_ip_free_monthly</code></td>
<td>the amount of IP address users can request for free per month (IP/hour)</td>
</tr>
<tr>
<td><code>limit_data_sent_free_monthly</code></td>
<td>the amount of data sent users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td><code>limit_data_received_free_monthly</code></td>
<td>the amount of data received users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td><code>price_rate_on</code></td>
<td>the price per Mbps of port speed per hour, charged for powered on VSs which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><code>price_rate_off</code></td>
<td>the price per Mbps of port speed per hour, charged for powered off VSs which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><code>price_ip_on</code></td>
<td>the price per IP address per hour, charged for powered on VSs which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td><code>price_ip_off</code></td>
<td>the price per IP address per hour, charged for powered</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>off VSs which are built in this network zone under this bucket (IP/hour)</td>
<td></td>
</tr>
<tr>
<td>price_data_sent</td>
<td>the price per GB of data sent per hour, charged for VSs which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_received</td>
<td>the price per GB of data received per hour, charged for VSs which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>backup_server_zone_resource</td>
<td>limit_backup_free</td>
</tr>
<tr>
<td>limit_backup_disk_size_free</td>
<td>the amount of disk space users can request for free to store their backups in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>limit_template_free</td>
<td>the amount of templates users can store in this backup server zone for free under this bucket (template/hour)</td>
</tr>
<tr>
<td>limit_template_disk_size_free</td>
<td>the amount of disk space users can request for free to store their templates in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>limit_ova_free</td>
<td>the amount of OVA users can store in this backup server zone for free under this bucket (OVA/hour)</td>
</tr>
<tr>
<td>limit_ova_disk_size_free</td>
<td>the amount of disk space users can</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>request for free to store their OVAs in this backup server zone under this bucket (GB/hour)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>price_backup</strong></td>
<td>the price per backup per hour, charged for the backups stored in this backup server zone under this bucket (backup/hour)</td>
</tr>
<tr>
<td><strong>price_backup_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user's backups stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_template</strong></td>
<td>the price per template per hour, charged for the backups stored on this backup server zone under this bucket (template/hour)</td>
</tr>
<tr>
<td><strong>price_template_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user's templates stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_ova</strong></td>
<td>the price per OVA per hour, charged for the backups stored on this backup server zone under this bucket (OVA/hour)</td>
</tr>
<tr>
<td><strong>price_ova_disk_size</strong></td>
<td>the price per GB per hour, charged for the disk size occupied by the user's OVAs stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>draas_resource</strong></td>
<td></td>
</tr>
<tr>
<td><strong>price_disk_size</strong></td>
<td>the additional price for disk size that applies to a virtual server with enabled DRaaS (GB/hour)</td>
</tr>
</tbody>
</table>
### Virtual Server

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>price_memory</code></td>
<td>the additional price for RAM that applies to a virtual server with enabled DRaaS (GB/hour)</td>
</tr>
<tr>
<td><code>price_cpus</code></td>
<td>the additional price for CPU that applies to a virtual server with enabled DRaaS (core/hour)</td>
</tr>
<tr>
<td><code>price_cpu_shares</code></td>
<td>the additional price for CPU shares that applies to a virtual server with enabled DRaaS (%/hour)</td>
</tr>
<tr>
<td><code>price_cpu_units</code></td>
<td>the additional price for CPU units that applies to a virtual server with enabled DRaaS (unit/hour)</td>
</tr>
<tr>
<td><code>price_nodes</code></td>
<td>the additional price for nodes that applies to a virtual server with enabled DRaaS (node/hour)</td>
</tr>
<tr>
<td><code>compute_resource_storing_resource limit_free</code></td>
<td>the amount of free disk space users can allocate to storing backups, ISOs and templates together (GB/hour)</td>
</tr>
<tr>
<td><code>price_backups_resource limit_free</code></td>
<td>the number of backups users can create for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td><code>price</code></td>
<td>the price per GB per hour of disk space the user allocates to storing backups, ISOs and templates (GB/hour)</td>
</tr>
<tr>
<td><code>templates_resource limit_free</code></td>
<td>the number of templates a user under this bucket can create for free (template/hour)</td>
</tr>
<tr>
<td>Resource</td>
<td>Limit</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>Virtual Server</td>
<td></td>
</tr>
<tr>
<td>price</td>
<td></td>
</tr>
<tr>
<td>iso_templates_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td>price</td>
<td></td>
</tr>
<tr>
<td>accelerated_servers_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td>price</td>
<td></td>
</tr>
<tr>
<td>autoscaled_servers_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td>price</td>
<td></td>
</tr>
<tr>
<td>solidfire_data_store_zone_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td>price_on</td>
<td></td>
</tr>
<tr>
<td>price_off</td>
<td></td>
</tr>
</tbody>
</table>
## Virtual Server

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>preconfigured_servers_resource</code></td>
<td>built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><code>price_on</code></td>
<td>the price per instance package per hour, charged for powered on VSs which are built on this instance package under this bucket</td>
</tr>
<tr>
<td><code>price_off</code></td>
<td>the price per instance package per hour, charged for powered off VSs which are built on this instance package under this bucket</td>
</tr>
<tr>
<td><code>price_overused_bandwidth</code></td>
<td>the price per overused bandwidth per hour (GB/hr), charged for all VSs which are built on this instance package under this bucket</td>
</tr>
</tbody>
</table>

### Page History

**v. 6.2 Edge 1**
- removed the `apply_to_all_resources_in_the_bucket` parameter

**v. 5.9**
- added the `apply_to_all_resources_in_the_bucket` parameter

**v. 5.7**
- added the `preconfigured_servers_resource` parameter that can have the following parameters:
  - `price_on`
  - `price_off`
  - `price_overused_bandwidth`

### 22.8.8 Add Rate Cards for Other Server Type

To add rate cards, use the following request:

POST /billing/buckets/:bucket_id/rate_cards.xml

POST /billing/buckets/:bucket_id/rate_cards.json

**XML Request Example**
curl -i -X POST -u user:userpass --url
' <rate_card><target_id>8</target_id><type>service_addon_resource</type><bucket_id>321</bucket_id><server_type>other</server_type><prices><price>12</price><price_cpu>23</price_cpu><price_memory>34</price_memory><price_disk_size>45</price_disk_size></prices></rate_card>'

Where:

bucket_id* - the ID of the bucket with which this rate card is associated.

server_type* - the server type this rate card is applicable to. In this case other.

target_id - the ID of the resource for which the prices are set.

type* - the type of the resource that is added to the rate card, it can be one of the following values:

- backup_resource_zone_resource
- edge_groups_resource
- template_resource
- service_addon_resource

prices - the array of prices and free limits for the resource that can be the following.

<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backup_resource_zone_resource</td>
<td>price_recovery_point_size, price_space_used, limit_free, limit_recovery_point_size_free, limit_space_used_free</td>
<td>- The price for a recovery point (backup) per hour.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The price for a recovery point (backup) size in Gb per hour.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The price for a total disk size (Gb/hour) taken by all backups on a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>particular virtual server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The number of recovery points (backup/hour) users can store in a backup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resource zone for free.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The size of recovery points (Gb/hour) users</td>
</tr>
<tr>
<td>Type</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>can consume in a backup resource zone for free.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The size of backups (Gb/hour) on a particular virtual server that users can consume in a backup resource zone for free.</td>
</tr>
<tr>
<td>edge_groups_resource</td>
<td>price</td>
<td>The price per GB of bandwidth.</td>
</tr>
<tr>
<td>service_addon_resource</td>
<td>price</td>
<td>The price per service add-on per hour.</td>
</tr>
<tr>
<td></td>
<td>price_cpu</td>
<td>The price for the CPU usage per hour per CPU core.</td>
</tr>
<tr>
<td></td>
<td>price_memory</td>
<td>The additional price for RAM per GB per hour.</td>
</tr>
<tr>
<td></td>
<td>price_disk_size</td>
<td>The additional price for disk size per GB per hour.</td>
</tr>
<tr>
<td>template_resource</td>
<td>price</td>
<td>The price per template in a template store.</td>
</tr>
</tbody>
</table>

**Page History**

v. 6.1 Edge 2
- added the following parameters for a backup_resource_zone_resource type:
  - price_space_used
  - limit_space_used_free

v. 6.0
- added the following parameters for a backup_resource_zone_resource type:
  - price_recovery_point_size
  - limit_recovery_point_size_free

v. 5.8
- added the backup_resource_zone_resource type

### 22.8.9 Edit Rate Cards for Baremetal Server Type

To edit rate cards, use the following request:

PUT /billing/buckets/:bucket_id/rate_cards.xml

PUT /billing/buckets/:bucket_id/rate_cards.json

XML Request Example
curl "http://onapp.test/billing/buckets/35/rate_cards.xml" \
-d '<rate_card> \
		<target_id>67</target_id> \
		<type>network_zone_resource</type> \
		<bucket_id>35</bucket_id> \
		<server_type>baremetal</server_type> \
		<prices> \
			<limit_ip_free>12</limit_ip_free> \
			<price_ip>23</price_ip> \
		</prices> \
	</rate_card>' \
-X PUT \
-u user:userpass \
-H "Accept: application/xml" \
-H "Content-Type: application/xml"

JSON Request Example

```bash
-u user:userpass \
-H "Accept: application/json" \
-H "Content-Type: application/json"
```

Where:

- **target_id** - the ID of the resource for which the prices are set.
- **type** - the type of the resource for which configuration is set. The value can be network_zone_resource.
- **bucket_id** - the ID of the bucket with which this rate card is associated.
- **server_type** - the server type this rate card is applicable to.
- **prices** - set the price and free limits for network zone resources:
  - **limit_ip_free** - the number of IP addresses users can request for free either per hour or per month
  - **price_ip** - the price per IP address per hour charged for VSs which are built in this network zone under this bucket

Page History

v. 6.2 Edge 1
- removed the `apply_to_all_resources_in_the_bucket` parameter

v. 5.9
- added the `apply_to_all_resources_in_the_bucket` parameter

**22.8.10 Edit Rate Cards for Smart Server Type**

To edit rate cards for smart server type, use the following request:

```
PUT /billing/buckets/:bucket_id/rate_cards.xml
```
PUT /billing/buckets/:bucket_id/rate_cards.json

**XML Request Example**

curl -X PUT http://onapp.test/billing/buckets/5/rate_cards.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml' -u user:userpass -d '<rate_card><bucket_id type="integer">5</bucket_id><server_type>smart</server_type><target_id type="integer">7</target_id><type>network_zone_resource</type><timing_strategy>hourly</timing_strategy><target_name>Smart Network Zone</target_name><prices><price_data_sent type="decimal">10.0</price_data_sent><price_data_received type="decimal">1.0</price_data_received></prices></rate_card>'

**JSON Request Example**

curl -X PUT http://onapp.test/billing/buckets/5/rate_cards.json -H 'Accept: application/json' -H 'Content-Type: application/json' -u user:userpass -d '{"bucket_id": 5, "server_type": "smart", "target_id": 7, "type": "network_zone_resource", "timing_strategy": "hourly", "target_name": "Smart Network Zone", "prices": {"price_data_sent type": 10.0, "price_data_received": 1.0} }'

Where:

- **bucket_id** - the ID of the bucket with which this rate card is associated
- **server_type** - the server type this rate card is applicable to (*smart* for Smart server type)
- **target_id** - the ID of the resource that is added to the rate card
- **type** - the type of the resource that is added to the rate card, it can be one of the following values:
  - backups_resource
  - backup_server_zone_resource
  - compute_resource_storing_resource
  - compute_zone_resource
  - data_store_zone_resource
  - network_zone_resource
  - smart_servers_resource
- **timing_strategy** - the type of billing for each resource (*hourly* for Smart server type)
- **target_name** - the name of the resource that is added to the rate card. For example, it can be a label of a network or compute zone, etc.
- **prices** - the array of resource prices and free limits

<table>
<thead>
<tr>
<th>Smart Server Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backups_resource</td>
<td>limit_free</td>
<td>the number of backups users can create for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td>Smart Server Type</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>price</td>
<td>the price per backup created by the user under this bucket per hour (backup/hour)</td>
<td></td>
</tr>
<tr>
<td>backup_server_zone_resource</td>
<td>limit_backup_free</td>
<td>the amount of backups users can store in this backup server zone for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td>limit_template_free</td>
<td>the amount of templates users can store in this backup server zone for free under this bucket (template/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_backup_disk_size_free</td>
<td>the amount of disk space users can request for free to store their backups in this backup server zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_template_disk_size_free</td>
<td>the amount of disk space users can request for free to store their templates in this backup server zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>price_backup</td>
<td>the price per backup per hour, charged for the backups stored in this backup server zone under this bucket (backup/hour)</td>
<td></td>
</tr>
<tr>
<td>price_backup_disk_size</td>
<td>the price per GB per hour, charged for the disk size occupied by the user's backups stored in this backup server zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>price_template</td>
<td>the price per template per hour, charged for the backups stored on this backup server zone under this bucket (template/hour)</td>
<td></td>
</tr>
<tr>
<td>price_template_disk_size</td>
<td>the price per GB per hour, charged for the disk size occupied by the user's templates stored in this backup server zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>compute_resource_storing_resource</td>
<td>limit_free</td>
<td>the amount of free disk space users can allocate to storing backups, ISOs and</td>
</tr>
</tbody>
</table>
### Smart Server Type

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>templates together</td>
<td>the price per GB per hour of disk space the user allocates to storing backups, ISOs, and templates (GB/hour)</td>
</tr>
<tr>
<td>price</td>
<td>the amount of CPU cores users can request for free for the total number of smart servers built in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td>limit_free_cpu</td>
<td>the amount of CPU shares users can request for free for the total number of their smart servers built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td>limit_free_cpu_share</td>
<td>the amount of CPU units users can request for free for the total number of their smart servers built in this compute zone under this bucket (unit/hour)</td>
</tr>
<tr>
<td>limit_free_memory</td>
<td>the amount of RAM users can request for free for the total number of their smart servers built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_on_cpu</td>
<td>the price per CPU core per hour, charged for powered on smart servers which are built in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td>price_off_cpu</td>
<td>the price per CPU core per hour, charged for powered off smart servers which are built in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td>price_on_cpu_share</td>
<td>the price for CPU shares, charged for powered on smart servers which are built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td>price_off_cpu_share</td>
<td>the price for CPU shares, charged for powered off smart servers which are built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td><strong>Smart Server Type</strong></td>
<td></td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>built in this compute zone under this bucket:</strong> (%/hour)</td>
<td></td>
</tr>
<tr>
<td><strong>price_on_cpu_units</strong></td>
<td>the price per CPU unit per hour, charged for powered on smart servers which are built in this compute zone under this bucket (CPU unit/hour)</td>
</tr>
<tr>
<td><strong>price_off_cpu_units</strong></td>
<td>the price per CPU unit per hour, charged for powered off smart servers which are built in this compute zone under this bucket (CPU unit/hour)</td>
</tr>
<tr>
<td><strong>price_on_memory</strong></td>
<td>the price for RAM, charged for powered on smart servers which are built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_off_memory</strong></td>
<td>the price for RAM, charged for powered off smart servers which are built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>data_store_zone_resource</strong></td>
<td></td>
</tr>
<tr>
<td><strong>limit_free</strong></td>
<td>the amount of disk space users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td><strong>limit_data_read_free</strong></td>
<td>the amount of read data users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td><strong>limit_data_written_free</strong></td>
<td>the amount of written data users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td><strong>limit_reads_completed_free</strong></td>
<td>the amount of input requests users can request for free per hour (1M requests/hour)</td>
</tr>
<tr>
<td><strong>limit_writes_completed_free</strong></td>
<td>the amount of output requests users can request for free per hour (1M requests/hour)</td>
</tr>
<tr>
<td><strong>price_on</strong></td>
<td>the price per GB of disk space per hour, charged for powered on smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><strong>price_off</strong></td>
<td>the price per GB of disk space per hour, charged</td>
</tr>
<tr>
<td>Smart Server Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>for powered off smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_read</td>
<td>the price per GB of read data per hour, charged for smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_written</td>
<td>the price per GB of written data per hour, charged for smart servers which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_reads_completed</td>
<td>the price per 1M input requests per hour, charged for smart servers which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>price_writes_completed</td>
<td>the price per 1M output requests per hour, charged for smart servers which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>network_zone_resource</td>
<td>limit_rate_free the amount of port speed users can request for free for the total number of their smart servers built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_ip_free the amount of IP addresses users can request for free per hour (IP/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_data_sent_free the amount of data sent users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>limit_data_received_free the amount of data received users can request for free per hour (GB/hour)</td>
</tr>
<tr>
<td></td>
<td>price_rate_on the price per Mbps of port speed per hour, charged for powered on smart servers which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
</tbody>
</table>
### Smart Server Type

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>price_rate_off</td>
<td>the price per Mbps of port speed per hour, charged for powered off smart servers which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td>price_ip_on</td>
<td>the price per IP address per hour, charged for powered on smart servers which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td>price_ip_off</td>
<td>the price per IP address per hour, charged for powered off smart servers which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td>price_data_sent</td>
<td>the price per GB of data sent per hour, charged for smart servers which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_received</td>
<td>the price per GB of data received per hour, charged for smart servers which are built in this network zone under this bucket (GB/hour)</td>
</tr>
</tbody>
</table>

### Page History

- **v. 6.2 Edge 1**
  - removed the `apply_to_all_resources_in_the_bucket` parameter

- **v. 5.9**
  - added the `apply_to_all_resources_in_the_bucket` parameter

### 22.8.11 Edit Rate Cards for Virtual Server Type

To edit rate cards, use the following request:

PUT /billing/buckets/:bucket_id/rate_cards.xml

PUT /billing/buckets/:bucket_id/rate_cards.json

XML Request Example
curl "http://onapp.test/billing/buckets/2/rate_cards.xml" -d
  '<rate_card>
    <bucket_id type="integer">2</bucket_id>
    <server_type>virtual</server_type>
    <target_id type="integer">12</target_id>
    <type<data_store_zone_resource</type>
    <timing_strategy>hourly</timing_strategy>
    <target_name>Default DataStore Zone</target_name>
    <prices>
      <limit_free type="decimal">10.0</limit_free>
      <price_on type="decimal">2.0</price_on>
      <price_off type="decimal">1.0</price_off>
    </prices>
  </rate_card>'
-X PUT
-u user:userpass
-H "Accept: application/xml"
-H "Content-Type: application/xml"

JSON Request Example

curl "http://onapp.test/billing/buckets/2/rate_cards.json" -d
  '{"bucket_id": 2,"server_type": "virtual","target_id": 12,"type":"data_store_zone_resource","timing_strategy": "hourly","prices":{"limit_free": 10,"price_on": 2,"price_off": 1}}'
-X PUT
-u user:userpass
-H "Accept: application/json"
-H "Content-Type: application/json"

Where:

bucket_id - the ID of the bucket with which this rate card is associated
server_type - the server type this rate card is applicable to (virtual for Virtual server type)
target_id - the ID of the resource that is added to the rate card
type - the type of the resource that is added to the rate card, it can be one of the following values:
- network_zone_resource
- compute_zone_resource
- backup_server_zone_resource
- solidfire_data_store_zone_resource
- autoscaled_servers_resource
- templates_resource
- compute_resource_storing_resource
- backups_resource
- `iso_templates_resource`

- `accelerated_servers_resource`

- `data_store_zone_resource`

- `preconfigured_servers_resource`

  `timing_strategy` - the type of billing for each resource: *hourly* or *monthly* (on peak usage)

  `target_name` - the name of the resource that is added to the rate card. For example, it can be a label of a network or *compute* zone, etc.

  `prices` - the array of resource prices and free limits

---

### Virtual Server

<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>compute_zone_resource</code></td>
<td><code>limit_free_cpu</code></td>
<td>the amount of CPU cores users can request for free for the total number of VSs built in this compute zone under this bucket (CPU core/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_cpu_share</code></td>
<td>the amount of CPU shares users can request for free for the total number of their VSs built in this compute zone under this bucket (%/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_cpu_units</code></td>
<td>the amount of CPU units users can request for free for the total number of their VSs built in this compute zone under this bucket (unit/hour)</td>
</tr>
<tr>
<td></td>
<td><code>limit_free_memory</code></td>
<td>the amount of RAM users can request for free for the total number of their VSs built in this compute zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td></td>
<td><code>price_on_cpu</code></td>
<td>the price per CPU core per hour, charged for powered on VSs which are built in this compute zone under this bucket</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>price_off_cpu</td>
<td>the price per CPU core per hour, charged for powered off VSs which are built in this compute zone under this bucket (CPU core/hour)</td>
<td></td>
</tr>
<tr>
<td>price_on_cpu_share</td>
<td>the price for CPU shares, charged for powered on VSs which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off_cpu_share</td>
<td>the price for CPU shares, charged for powered off VSs which are built in this compute zone under this bucket (%/hour)</td>
<td></td>
</tr>
<tr>
<td>price_on_cpu_units</td>
<td>the price per CPU unit per hour, charged for powered on VSs which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off_cpu_units</td>
<td>the price per CPU unit per hour, charged for powered off VSs which are built in this compute zone under this bucket (CPU unit/hour)</td>
<td></td>
</tr>
<tr>
<td>price_on_memory</td>
<td>the price for RAM, charged for powered on VSs which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off_memory</td>
<td>the price for RAM, charged for powered off VSs which are built in this compute zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>data_store_zone_resource</td>
<td>limit_free</td>
<td></td>
</tr>
<tr>
<td>data_store_zone_resource</td>
<td>the amount of disk space users can</td>
<td></td>
</tr>
<tr>
<td>Virtual Server</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>limit_data_read_free</td>
<td>the amount of read data users can request for free either per hour or per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_data_written_free</td>
<td>the amount of written data users can request for free either per hour or per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_reads_completed_free</td>
<td>the amount of input requests users can request for free either per hour or per month (1M requests/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_writes_completed_free</td>
<td>the amount of output requests per hour users can request for free (1M requests/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_free_monthly</td>
<td>the amount of disk space users can request for free per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_data_read_free_monthly</td>
<td>the amount of read data users can request for free per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_data_written_free_monthly</td>
<td>the amount of written data users can request for free per month (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_reads_completed_free_monthly</td>
<td>the amount of input requests users can request for free per month (1M requests/hour)</td>
<td></td>
</tr>
<tr>
<td>limit_writes_completed_free_monthly</td>
<td>the amount of output requests users can request for free either per hour or per month (1M requests/hour)</td>
<td></td>
</tr>
</tbody>
</table>
| price_on | the price per GB of disk space per hour,
<table>
<thead>
<tr>
<th>Virtual Server</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>charged for powered on VSs which are built in this data store zone under this bucket (GB/hour)</td>
<td></td>
</tr>
<tr>
<td>price_off</td>
<td>the price per GB of disk space per hour, charged for powered off VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_read</td>
<td>the price per GB of read data per hour, charged for VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_written</td>
<td>the price per GB of written data per hour, charged for VSs which are built in this data store zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_reads_completed</td>
<td>the price per 1M input requests per hour, charged for VSs which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>price_writes_completed</td>
<td>the price per 1M output requests per hour, charged for VSs which are built in this data store zone under this bucket (1M requests/hour)</td>
</tr>
<tr>
<td>network_zone_resource</td>
<td>limit_rate_free</td>
</tr>
<tr>
<td>limit_ip_free</td>
<td>the amount of IP address users can request for free</td>
</tr>
<tr>
<td><strong>Virtual Server</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------</td>
</tr>
<tr>
<td><strong>limit_data_sent_free</strong></td>
<td>the amount of data sent users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td><strong>limit_data_received_free</strong></td>
<td>the amount of data received users can request for free either per hour or per month (GB/hour)</td>
</tr>
<tr>
<td><strong>limit_ip_free_monthly</strong></td>
<td>the amount of IP address users can request for free per month (IP/hour)</td>
</tr>
<tr>
<td><strong>limit_data_sent_free_monthly</strong></td>
<td>the amount of data sent users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td><strong>limit_data_received_free_monthly</strong></td>
<td>the amount of data received users can request for free per month (GB/hour)</td>
</tr>
<tr>
<td><strong>price_rate_on</strong></td>
<td>the price per Mbps of port speed per hour, charged for powered on VSs which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><strong>price_rate_off</strong></td>
<td>the price per Mbps of port speed per hour, charged for powered off VSs which are built in this network zone under this bucket (Mbps/hour)</td>
</tr>
<tr>
<td><strong>price_ip_on</strong></td>
<td>the price per IP address per hour, charged for powered on VSs which are built in this network zone under this bucket (IP/hour)</td>
</tr>
<tr>
<td><strong>price_ip_off</strong></td>
<td>the price per IP address per hour, charged for powered off VSs which are built in this network</td>
</tr>
</tbody>
</table>
### Virtual Server

<table>
<thead>
<tr>
<th>Virtual Server</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zone under this bucket (IP/hour)</td>
<td>the price per GB of data sent per hour, charged for VSs which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_sent</td>
<td>the price per GB of data received per hour, charged for VSs which are built in this network zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_data_received</td>
<td>the amount of backups users can store in this backup server zone for free under this bucket (backup/hour)</td>
</tr>
<tr>
<td>limit_backup_free</td>
<td>the amount of disk space users can request for free to store their backups in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>limit_backup_disk_size_free</td>
<td>the amount of templates users can store in this backup server zone for free under this bucket (template/hour)</td>
</tr>
<tr>
<td>limit_template_free</td>
<td>the amount of disk space users can request for free to store their templates in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>limit_template_disk_size_free</td>
<td>the amount of OVA users can store in this backup server zone for free under this bucket (OVA/hour)</td>
</tr>
<tr>
<td>limit_ova_free</td>
<td>the amount of disk space users can request for free to store their OVAs in</td>
</tr>
<tr>
<td>Virtual Server</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>price_backup</td>
<td>the price per backup per hour, charged for the backups stored in this backup server zone under this bucket (backup/hour)</td>
</tr>
<tr>
<td>price_backup_disk_size</td>
<td>the price per GB per hour, charged for the disk size occupied by the user's backups stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_template</td>
<td>the price per template per hour, charged for the backups stored on this backup server zone under this bucket (template/hour)</td>
</tr>
<tr>
<td>price_template_disk_size</td>
<td>the price per GB per hour, charged for the disk size occupied by the user's templates stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>price_ova</td>
<td>the price per OVA per hour, charged for the backups stored on this backup server zone under this bucket (OVA/hour)</td>
</tr>
<tr>
<td>price_ova_disk_size</td>
<td>the price per GB per hour, charged for the disk size occupied by the user's OVAs stored in this backup server zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td>draas_resource</td>
<td>the additional price for disk size that applies to a virtual server with enabled DRaaS (GB/hour)</td>
</tr>
<tr>
<td>price_disk_size</td>
<td>the additional price for disk size that applies to a virtual server</td>
</tr>
<tr>
<td>price_memory</td>
<td>the additional price for RAM that applies to a virtual server</td>
</tr>
<tr>
<td><strong>Virtual Server</strong></td>
<td><strong>with enabled DRaaS (GB/hour)</strong></td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td><strong>price_cpu</strong></td>
<td>the additional price for CPU that applies to a virtual server with enabled DRaaS (core/hour)</td>
</tr>
<tr>
<td><strong>price_cpu_shares</strong></td>
<td>the additional price for CPU shares that applies to a virtual server with enabled DRaaS (%/hour)</td>
</tr>
<tr>
<td><strong>price_cpu_units</strong></td>
<td>the additional price for CPU units that apply to a virtual server with enabled DRaaS (unit/hour)</td>
</tr>
<tr>
<td><strong>price_nodes</strong></td>
<td>the additional price for nodes that apply to a virtual server with enabled DRaaS (node/hour)</td>
</tr>
<tr>
<td><strong>compute_resource_storing_resource</strong></td>
<td><strong>limit_free</strong></td>
</tr>
<tr>
<td><strong>price</strong></td>
<td>the price per GB per hour of disk space the user allocates to storing backups, ISOs and templates (GB/hour)</td>
</tr>
<tr>
<td><strong>backups_resource</strong></td>
<td><strong>limit_free</strong></td>
</tr>
<tr>
<td><strong>price</strong></td>
<td>the price per backup created by the user under this bucket per hour (backup/hour)</td>
</tr>
<tr>
<td><strong>templates_resource</strong></td>
<td><strong>limit_free</strong></td>
</tr>
<tr>
<td><strong>price</strong></td>
<td>the price per template created by</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Limit Type</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>iso_templates_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>accelerated_servers_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>autoscaled_servers_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>solidfire_data_store_zone_resource</td>
<td>limit_free</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| iso_templates_resource              | limit_free | the user under this bucket per hour (template/hour)                         |
|                                     |            | price                                                                       |
| accelerated_servers_resource        | limit_free | the amount of virtual servers for which the user can enable acceleration for free under this bucket (VS/hour) |
|                                     |            | price                                                                       |
| autoscaled_servers_resource         | limit_free | the amount of virtual servers for which the user can enable autoscaling for free under this bucket (VS/hour) |
|                                     |            | price                                                                       |
| solidfire_data_store_zone_resource  | limit_free | the number of IOPS requests a user under this bucket can create for free (IOPS/hour) |
|                                     |            | price_on                                                                    |
|                                     |            | price_off                                                                   |
### Virtual Server

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>preconfigured_servers_resource</code></td>
<td>zone under this bucket (GB/hour)</td>
</tr>
<tr>
<td><code>price_on</code></td>
<td>the price per instance package per hour, charged for powered on VSs which are built on this instance package under this bucket</td>
</tr>
<tr>
<td><code>price_off</code></td>
<td>the price per instance package per hour, charged for powered off VSs which are built on this instance package under this bucket</td>
</tr>
<tr>
<td><code>price_overused_bandwidth</code></td>
<td>the price per overused bandwidth per hour (GB/hr), charged for all VSs which are built on this instance package under this bucket</td>
</tr>
</tbody>
</table>

### Page History

**v. 6.2 Edge 1**
- removed the `apply_to_all_resources_in_the_bucket` parameter

**v. 5.9**
- added the `apply_to_all_resources_in_the_bucket` parameter

**v. 5.7**
- added the `preconfigured_servers_resource` parameter that can have the following parameters:
  - `price_on`
  - `price_off`
  - `price_overused_bandwidth`

### 22.8.12 Edit Rate Cards for Other Server Type

To edit rate cards, use the following request:

PUT /billing/buckets/:bucket_id/rate_cards.xml

PUT /billing/buckets/:bucket_id/rate_cards.json

**XML Request Example**
JSON Request Example

```bash
curl -X PUT -u user:userpass --url
'{"rate_card": {
  "target_id": 7, "type": "edge_groups_resource", "bucket_id": 367, "server_type": "other", "prices": {
    "price": 0, "limit_free": -2.22
  }
}}'
```

**Where:**

- **bucket_id** - the ID of the bucket with which this rate card is associated
- **server_type** - the server type this rate card is applicable to, in this case **other**
- **target_id** - the ID of the resource that is added to the rate card
- **type** - the type of the resource that is added to the rate card, it can be one of the following values:
  - **backup_resource_zone_resource**
  - **edge_groups_resource**
  - **template_resource**
  - **service_addon_resource**
- **prices** - the array of prices and free limits for the resource that can be the following:

<table>
<thead>
<tr>
<th>Type</th>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backup_resource_zone_resource</td>
<td>price, price_recovery_point_size, price_space_used, limit_free, limit_recovery_point_size_free, limit_space_used_free</td>
<td>• The price for a recovery point (backup) per hour.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The price for a recovery point (backup) size in Gb per hour.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The price for a total disk size (Gb/hour) taken by all backups on a</td>
</tr>
<tr>
<td></td>
<td></td>
<td>particular virtual server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The number of recovery points (backup/hour) users can store in a backup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resource zone for free.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The size of recovery points (Gb/hour) users can consume in a backup</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resource zone for free.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The size of backups (Gb/hour) on a particular</td>
</tr>
<tr>
<td>Type</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>virtual server that users can consume in a backup resource zone for free.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>edge_groups_resource</td>
<td>price</td>
<td>The price per GB of bandwidth.</td>
</tr>
<tr>
<td>service_addon_resource</td>
<td>price</td>
<td>The price per service add-on per hour.</td>
</tr>
<tr>
<td>price_cpu</td>
<td></td>
<td>The price for the CPU usage per hour per CPU core.</td>
</tr>
<tr>
<td>price_memory</td>
<td></td>
<td>The additional price for RAM per GB per hour.</td>
</tr>
<tr>
<td>price_disk_size</td>
<td></td>
<td>The additional price for disk size per GB per hour.</td>
</tr>
<tr>
<td>template_resource</td>
<td>price</td>
<td>The price per template in a template store.</td>
</tr>
</tbody>
</table>

**Page History**

v. 6.1 Edge 2
- added the following parameters for a backup_resource_zone_resource type:
  - price_space_used
  - limit_space_used_free

v. 6.0
- added the following parameters for a backup_resource_zone_resource type:
  - price_recovery_point_size
  - limit_recovery_point_size_free

v. 5.8
- added the backup_resource_zone_resource type

### 22.8.13 Delete Resources from Rate Cards for Baremetal Server Type

If you remove a compute/data store/network/backup server zone from the Rate Card, the prices for the removed resource will be set to zero for the servers using this zone(s).

To delete resources from rate cards, use the following request:

DELETE /billing/buckets/:bucket_id/rate_cards.xml
DELETE /billing/buckets/:bucket_id/rate_cards.json

**XML Request Example**
22.8.14 Delete Resources from Rate Cards for Smart Server Type

If you remove a compute/data store/network/backup server zone from the Rate Card, the prices for the removed resource will be set to zero for the servers using this zone(s).

To delete resources from rate cards, use the following request:

DELETE /billing/buckets/:bucket_id/rate_cards.xml
DELETE /billing/buckets/:bucket_id/rate_cards.json

XML Request Example

curl -i -X DELETE -u user:userpass --url
http://onapp.test/billing/buckets/5/rate_cards/delete.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'<rate_card><type>compute_zone_resource</type><bucket_id>5</bucket_id><server_type>smart</server_type><target_id>10</target_id></rate_card>'

JSON Request Example

curl -i -X DELETE -u user:userpass --url
http://onapp.test/billing/buckets/5/rate_cards/delete.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d '{"rate_card": {
"type": "compute_zone_resource", "bucket_id": 5, "server_type":
"smart"}}'
curl -i -X DELETE

Where:

type - the type of the resource that is deleted from the rate card, it can be one of the following values:

- backups_resource
- backup_server_zone_resource
- compute_resource_storing_resource
- compute_zone_resource
- data_store_zone_resource
- network_zone_resource
- smart_servers_resource

bucket_id - the ID of the bucket with which this rate card is associated
server_type - the server type this rate card is applicable to (smart for Smart server type)
target_id - the ID of the resource that is deleted from the rate card

22.8.15 Delete Resources from Rate Cards for Virtual Server Type

If you remove a compute/data store/network/backup server zone from the Rate Card, the prices for the removed resource will be set to zero for the servers using this zone(s).

To delete resources from rate cards, use the following request:

DELETE /billing/buckets/:bucket_id/rate_cards.xml
DELETE /billing/buckets/:bucket_id/rate_cards.json

XML Request Example

curl -i -X DELETE -u user:userpass --url

JSON Request Example
Where:

type - the type of the resource that is deleted from the rate card, it can be one of the following values:

- network_zone_resource
- compute_zone_resource
- backup_server_zone_resource
- solidfire_data_store_zone_resource
- autoscaled_servers_resource
- templates_resource
- compute_resource_storing_resource
- backups_resource
- iso_templates_resource
- accelerated_servers_resource
- data_store_zone_resource
- preconfigured_servers_resource

bucket_id - the ID of the bucket with which this rate card is associated

server_type - the server type this rate card is applicable to (virtual for Virtual server type)

target_id - the ID of the resource that is deleted from the rate card

22.8.16 Delete Resource from Rate Card for Other Server Type

To delete resources from rate card, use the following request:

DELETE /billing/buckets/:bucket_id/rate_cards.xml
DELETE /billing/buckets/:bucket_id/rate_cards.json

XML Request Example
curl -i -X DELETE -u user:userpass --url
http://onapp.test/billing/buckets/308/rate_cards/delete.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml' -d '<rate_card><type>template_resource</type><bucket_id>308</bucket_id><server_type>other</server_type><target_id>1</target_id></rate_card>'

JSON Request Example

curl -i -X DELETE -u user:userpass --url
http://onapp.test/billing/buckets/308/rate_cards/delete.json -H 'Accept: application/json' -H 'Content-Type: application/json' -d '{"rate_card": {"type": "template_resource", "bucket_id": 308, "server_type": "other", "target_id": 1}}'

Where:

- **type** - the type of the resource for which configuration is set, it can be one of the following values:
  - `backup_resource_zone_resource`
  - `edge_groups_resource`
  - `template_resource`
  - `service_addon_resource`

- **bucket_id** - the ID of the bucket with which this access control is associated
- **server_type** - the server type this access control is applicable to, in this case, **other**
- **target_id** - the ID of the resource which is deleted

Page History

- v. 5.8
  - added the `backup_resource_zone_resource` type
23 Check Password Strength

To check password strength, use the following request:

GET http://onapp.test/password_strength_meter.xml?password=password_sample
GET http://onapp.test/password_strength_meter.json?password=password_sample

**XML Request Example**

```
curl -X GET -u user:userpass
http://onapp.test/password_strength_meter.xml?password=giey869$gj&HKJGY798
```

**JSON Request Example**

```
curl -X GET -u user:userpass
http://onapp.test/password_strength_meter.json?password=giey869$gj&HKJGY798
```

Where you have to specify your password instead of the `password_sample`
24 CloudBoot IP Addresses

CloudBoot IP addresses are IP addresses that are in use and are available for compute resource cloudbooting. The CloudBoot IP addresses are managed similarly to regular IPs used for VSs.

24.1 Get List of CloudBoot IP Addresses

To get the list of CloudBoot IP addresses, use the following request:

GET /cloud_boot_ip_addresses.xml
GET /cloud_boot_ip_addresses.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<ip_addresses type="array">
  <ip_address>
    <address>192.168.1.160</address>
    <broadcast>192.168.1.255</broadcast>
    <created_at type="datetime">2012-07-31T03:04:42-11:00</created_at>
    <customer_network_id nil="true"/>
    <disallowed_primary nil="true"/>
    <gateway>192.168.1.1</gateway>
    <hypervisor_id nil="true"/>
    <id type="integer">72</id>
    <ip_address_pool_id nil="true"/>
    <network_address>192.168.1.0</network_address>
    <network_id nil="true"/>
    <pxe type="boolean">true</pxe>
    <updated_at type="datetime">2012-07-31T03:04:42-11:00</updated_at>
    <user_id nil="true"/>
    <free type="boolean">true</free>
    <netmask>255.255.255.0</netmask>
  </ip_address>
  ...
</ip_addresses>
```

Where:

- **ip_addresses** - an array with all CloudBoot IP addresses
- **address** - IP address
- **broadcast** - broadcast address
- **created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
customer_network_id - the ID of the customer VLAN the IP address belongs to
disallowed_primary - true if not allowed to be used as primary, otherwise false
gateway - gateway address
hypervisor_id - the ID of a compute resource the IP address is associated with
id - the ID of the IP address
ip_address_pool_id - ID of the IP address pool the IP address is associated with
network_address - the address of the network
network_id - the ID of the network
pxe - true, if this compute resource address can be used for cloudbooting a compute resource
updated_at - the date when the network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
user_id - the ID of the user this IP address is assigned to
free - true if free, otherwise false
netmask - netmask for the IP address

24.2 Add CloudBoot IP Address

To add a CloudBoot IP address, use the following request:

POST /cloud_boot_ip_addresses.xml
POST /cloud_boot_ip_addresses.json

**XML Request Example**

```
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<ip_address><address>7.7.7.23</address></ip_address>' --url http://onapp.test/cloud_boot_ip_addresses.xml
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"ip_address":{"address":"7.7.7.23"}}' --url http://onapp.test/cloud_boot_ip_addresses.json
```

Where:

address - the IP address you want to add

**Page History**

v. 5.4

- removed netmask, broadcast, network_address, gateway
  and disallowed_primary parameters
24.3 Delete Cloud Boot IP Address

To delete a CloudBoot IP address, use the following request:

DELETE /cloud_boot_ip_addresses/:id.xml
DELETE /cloud_boot_ip_addresses/:id.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
http://onapp.test/cloud_boot_ip_addresses/14.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
http://onapp.test/cloud_boot_ip_addresses/14.json
```

Where you have to specify ID of a Cloud Boot IP address you want to delete.

**PLEASE NOTE:** You can't delete an IP address that is in use.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no CloudBoot IP address with a requested ID, or URL is incorrect.

24.4 Create CloudBoot IP Net

To create a CloudBoot IP Net, use the following request:

POST /settings/networks/network_id/ip_nets.xml
POST /settings/networks/network_id/ip_nets.json

**Where:**

*network_id* - ID of the CloudBoot network which is located in SystemNetworkGroup

**XML Request Example**

```bash
```

**JSON Request Example**

Where:

- **label** - the name of the IP net
- **add_default_ip_range** - set this parameter to `1` for the default IP range to be added to the IP net automatically. Otherwise, set `0`, then you'll need to add the required IP ranges after the IP net is created.
- **network_mask** - the network mask
- **network_address** - the network address of the IP net

### 24.5 Create CloudBoot IP Range

To create an IP range in a network, use the following request:

POST /settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges.xml

POST /settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges.json

Where:

- **network_id** - ID of the CloudBoot network which is located in SystemNetworkGroup

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:

- **end_address** - the IP address with which your IP range ends
- **default_gateway** - the default gateway for the IP range
- **start_address** - the IP address with which your IP range starts
25 Compute Resources

Compute resources provide hardware resources for virtual servers. A specific physical compute resource server supplies the CPU, RAM and storage capacity from the Data Stores attached to that compute resource. All API calls are available to this class.

25.1 Get List of Compute Resources

To get the list of compute resources, use the following request:
GET /settings/hypervisors.xml
GET /settings/hypervisors.json

XML Request Example

```
curl -i -u user:userpass -X GET http://onapp.test/settings/hypervisors.xml
```

JSON Request Example

```
curl -i -u user:userpass -X GET http://onapp.test/settings/hypervisors.json
```

XML Output Example
hypervisor – an array of all compute resources in the cloud and their details
backup - true, if the CloudBoot compute resource is used as a backup server. This parameter is for CloudBoot compute resources only. For other compute resource types the backup value is 0.

backup_ip_address - provisioning network IP address

blocked - true if the compute resource is blocked, otherwise false
built - true if the compute resource is built, otherwise false

called_in_at – the date when the compute resource was called in the [YYYY][MM][DD][hh][mm][ss]Z format
connection_options - an array of the following vCenter cluster parameters:
• login - vCenter login
• password - vCenter password
• cluster_name - vCenter cluster name
• distributed_virtual_switch_name - distributed virtual switch label
cpu_idle - time of CPU delay

cpu_mhz - CPU operating frequency

created_at – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
custom_config - custom commands that run when the compute resource is booted
disable_failover – true if compute resource failover is disabled, otherwise false.
disk_pcis - comma-separated list of hypervisor disk pcis
distro - distributive label

enabled - true, if the compute resource is enabled, otherwise false
failure_count – the number of failures
format_disks - true, if the compute resource's disks are formatted during creation, otherwise false

free_mem - free compute resource memory

host - host label

hypervisor_group_id - the ID of a compute zone to which this compute resource is attached
hypervisor_type - the compute resource type
id - the compute resource ID
ip_address – the compute resource IP address
label - the compute resource label

list_of_logical_volumes - an array of compute resource logical volumes
list_of_volume_groups - an array of compute resource volume groups

list_of_zombie_domains - an array of zombie virtual servers

locked - true if the compute resource is locked, otherwise false

mac - compute resource MAC address

machine - architecture type

mem_info - Xen compute resource Dom0 size. This parameter is for Xen compute resources only. For other compute resource types the mem_info value is 0.

mtu - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.

The maximum transportation unit (MTU) is the maximum size of a unit that can be transmitted transferred via ethernet traffic. Any data that exceed the specified MTU value will be divided into smaller units before being transferred. Utilization of jumbo frames allows to reduce increase throughput and increase CPU utilization during large size file transfers.

online - true if online, otherwise false

ovs - whether the OpenvSwitch is enabled or not. Redundant parameter.

passthrough_disks - true, if the disks are passed through to the storage nodes, otherwise 0

release - compute resource kernel version

segregation_os_type - an operating system type (can be Any OS, Windows only or Non-Windows)

server_type - server type:
- virtual
- smart
- baremetal

spare – true if no VSs are assigned, otherwise false

storage_channel - storage channel for the communication

threads_per_core - compute resource core and CPU ratio. For example, if threads_per_core = 2, compute resource with 2 cores will have 4 CPUs.

total_mem - total compute resource memory

total_zombie_mem - memory space occupied by zombie disks

updated_at – the date when the record was made in the DB in the [YYYY][MM][DD][hh][mm][ss]Z format

uptime - compute resource uptime value; shows how long the compute resource is online
vmware_total_cpu_cores - the total number of VMware compute resource CPU cores

total_cpus – the number of virtual cores

free_memory – free RAM (MB) of compute resource. It is calculated as follows: total memory - memory overhead
used_cpu_resources – the percentage of used CPU resources

total_memory – total RAM (MB) of compute resource
cpu_cores – the number of physical cores per compute resource

free_disk_space - free compute resource disk space in GB

memory_allocated_by_running_vms - the compute resource RAM in MB allocated to the virtual servers, which are currently running on this compute resource

total_memory_allocated_by_vms - the compute resource RAM in MB allocated to all virtual servers of this compute resource

cpu_units - the amount of CPU units assigned to this compute resource

cpu_flags - an array of CPU flags of the compute resource for which the extended CPU configuration is enabled

cpu_model - a CPU model of the compute resource for which the extended CPU configuration (flags) is enabled

*Memory Overhead for Compute Resources

Each compute resource has a reserved memory overhead value. This value is pre-configured by default in info_hub.yml.

For XEN compute resource:
memory_overhead = 1536

For KVM compute resource:
memory_overhead = 400 + 0.024 * total_memory

Page History
v.5.9
- added the following parameters:
  - segregation_os_type
  - failover_recipe_id

v.5.7
- added cpu_model parameter

v.4.2
- added cpu_flags parameter

v.3.3
- added cpu_units parameter

v.3.1
- added the following parameters:
  - cpu_idle
  - cpu_mhz
25.2 Get Compute Resource Details

To get compute resource details, use the following request:

GET /settings/hypervisors/:id.xml
GET /settings/hypervisors/:id.json

**XML Request Example**

```
curl -i -u user:userpass -X GET
http://onapp.test/settings/hypervisors/1.xml
```

**JSON Request Example**

```
curl -i -u user:userpass -X GET
http://onapp.test/settings/hypervisors/1.json
```

**XML Output Example**
<hypervisor>
  <allow_unsafe_assigned_interrupts type="boolean">false</allow_unsafe_assigned_interrupts>
  <backup type="boolean">false</backup>
  <backup_ip_address>/backup_ip_address>
  <blocked type="boolean">false</blocked>
  <built type="boolean">false</built>
  <called_in_at nil="true"/>
  <connection_options nil="true"/>
  <cpu_cores type="integer">4</cpu_cores>
  <cpu_idle type="integer">93</cpu_idle>
  <cpu_mhz>1995</cpu_mhz>
  <cpu_units type="integer">3000</cpu_units>
  <cpus type="integer">4</cpus>
  <created_at type="datetime">2014-01-14T15:43:03:00</created_at>
  <custom_config nil="true"/>
  <disable_failover type="boolean">false</disable_failover>
  <disks_per_storage_controller type="integer">4</disks_per_storage_controller>
  <distro>centos6</distro>
  <enabled type="boolean">true</enabled>
  <failover_recipe_id>get_if_config</failover_recipe_id>
  <failure_count type="integer">0</failure_count>
  <format_disks type="boolean">false</format_disks>
  <free_mem type="integer">1206</free_mem>
  <host nil="true"/>
  <host_id nil="true"/>
  <hypervisor_group_id type="integer">1</hypervisor_group_id>
  <hypervisor_type>kvm</hypervisor_type>
  <id type="integer">1</id>
  <infiniband_identifier nil="true"/>
  <ip_address>109.123.91.38</ip_address>
  <label>KVM C6 HV1</label>
  <list_of_logical_volumes>...</list_of_logical_volumes>
  <list_of_volume_groups>...</list_of_volume_groups>
  <list_of_zombie_domains></list_of_zombie_domains>
  <locked type="boolean">false</locked>
  <machine>x86_64</machine>
  <mem_info type="integer">0</mem_info>
  <mtu type="integer">1500</mtu>
  <online type="boolean">true</online>
  <ovs nil="true"/>
  <passthrough_disks type="boolean">false</passthrough_disks>
  <power_cycle_command>echo "I want to reboot! I'am LAZY!"</power_cycle_command>
  <release>2.6.32-431.5.1.el6.x86_64</release>
  <segregation_os_type>any</segregation_os_type>
  <server_type>virtual</server_type>
  <spare type="boolean">false</spare>
  <storage_controller_memory_size type="integer">640</storage_controller_memory_size>
  <threads_per_core type="integer">1</threads_per_core>
  <total_mem type="integer">15936</total_mem>
  <total_zombie_mem type="integer">0</total_zombie_mem>
  <updated_at type="datetime">2014-08-06T15:17:16+03:00</updated_at>
  <uptime>13:16:42 up 41 days, 23:20, 1 user, load average: 0.04, 0.05, 0.06</uptime>
  <cpu_model>Nehalem</cpu_model>
  <total_cpus type="integer">4</total_cpus>
  <free_memory type="integer">612</free_memory>
  <used_cpu_resources type="integer">880</used_cpu_resources>
  <total_memory type="integer">15936</total_memory>
  <cpu_cores type="integer">4</cpu_cores>
  <free_disk_space>
Where:

`hypervisor` – an array of compute resource details

`backup` - true, if the CloudBoot compute resource is used as a backup server. This parameter is for CloudBoot compute resources only. For other compute resource types the `backup` value is 0.

`backup_ip_address` - provisioning network IP address

`blocked` - true if the compute resource is blocked, otherwise false

`built` - true if the compute resource is built, otherwise false

`called_in_at` – the date when the compute resource was called in the [YYYY][MM][DD]T[hh][mm][ss]Z format

`connection_options` - an array of the following vCenter cluster parameters:

- `login` - vCenter login
- `password` - vCenter password
- `cluster_name` - vCenter cluster name
- `distributed_virtual_switch_name` - distributed virtual switch label

`cpu_flags` - an array of CPU flags of the compute resource for which the extended CPU configuration is enabled

`cpu_idle` - time of CPU delay

`cpu_mhz` - CPU operating frequency

`created_at` – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

`custom_config` - custom commands that run when the compute resource is booted

`disable_failover` – true if compute resource failover is disabled, otherwise false.

`distro` - distributive label

`enabled` - true, if the compute resource is enabled, otherwise false

`failover_recipe_id` - the ID of a recipe to run before the failover process

`failure_count` – the number of failures
format_disks - true, if the compute resource's disks are formatted during creation, otherwise false

free_mem - free compute resource memory

host - host label

hypervisor_group_id - the ID of a compute zone to which this compute resource is attached
hypervisor_type - the compute resource type
id - the compute resource ID
ip_address – the compute resource IP address
label - the compute resource label

list_of_logical_volumes - an array of compute resource logical volumes
list_of_volume_groups - an array of compute resource volume groups
list_of_zombie_domains - an array of zombie virtual servers

locked - true if the compute resource is locked, otherwise false
mac - compute resource MAC address
machine - architecture type

mem_info - Xen compute resource Dom0 size. This parameter is for Xen compute resources only. For other compute resource types the mem_info value is 0.

mtu - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.

The maximum transportation unit (MTU) is the maximum size of a unit that can be transmitted transferred via ethernet traffic. Any data that exceed the specified MTU value will be divided into smaller units before being transferred. Utilization of jumbo frames allows to reduce increase throughput and increase CPU utilization during large size file transfers.

online - true if online, otherwise false
ovs - whether the OpenvSwitch is enabled or not. Redundant parameter.

passthrough_disks - true, if the disks are passed through to the storage nodes, otherwise 0
release - compute resource kernel version
segregation_os_type - an operating system type (can be Any OS, Windows only or Non-Windows)

server_type - server type:
  • virtual
- **smart**
- **baremetal**

**spare** – true if no VSs are assigned, otherwise false

**storage_channel** - storage channel for the communication

**threads_per_core** - compute resource core and CPU ratio. For example, if threads_per_core = 2, compute resource with 2 cores will have 4 CPUs.

**total_mem** - total compute resource memory

**total_zombie_mem** - memory space occupied by zombie disks

**updated_at** – the date when the record was made in the DB in the [YYYY][MM][DD]T[hh][mm][ss]Z format

**uptime** - compute resource uptime value; shows how long the compute resource is online

**vmware_total_cpu_cores** - the total number of VMware compute resource CPU cores

**total_cpus** – the number of virtual cores

**free_memory** – free RAM (MB) of compute resource. It is calculated as follows: total memory - memory overhead* - zombie VSs - sum of all VSs

**used_cpu_resources** – the percentage of used CPU resources

**total_memory** – total RAM (MB) of compute resource

**cpu_cores** – the number of physical cores per compute resource

**free_disk_space** - free compute resource disk space in GB

**memory_allocated_by_running_vms** - the compute resource RAM in MB allocated to the virtual servers, which are currently running on this compute resource

**total_memory_allocated_by_vms** - the compute resource RAM in MB allocated to all virtual servers of this compute resource

**cpu_units** - the amount of cpu units assigned to a compute resource

**allow_unsafe_assigned_interrupts** - true if the compute resource does not support the interrupt remapping, otherwise set false.

With **allow_unsafe_assigned_interrupts** parameter enabled, smart server is exposed to the PCI passthrough MSI trap injection.

**cpus** - a parameter duplicating **total_cpus**; reserved for future use

**disks_per_storage_controller** - the number of disks per controller virtual server. You can set form 1 to 4 disks. By default, the controller virtual server is created per 4 disk drives

**infiniband_identifier** - the identifier of the InfiniBand interface. For CloudBoot Compute resources only.

**power_cycle_command** - arbitrary command string to be executed by IPMI from the CP server. This parameter is for CloudBoot Compute resources.

**cpu_model** - a CPU model of the compute resource for which the extended CPU configuration (flags) is enabled
*Memory Overhead for Compute Resources*

Each compute resource has a reserved memory overhead value. This value is pre-configured by default in info_hub.yml.

For XEN compute resource:
memory_overhead = 1536

For KVM compute resource:
memory_overhead = 400 + 0.024 * total_memory

**Page History**

v.5.9
- added the following parameters:
  - segregation_os_type
  - failover_recipe_id

v.5.7
- added the cpu_model parameter.

v.4.2
- added cpu_flags parameter

v.3.3
- added cpu_units parameter

v.3.1
- added the following parameters:
  - cpu_idle
  - cpu_mhz
  - list_of_logical_volumes
  - list_of_volume_groups
  - list_of_zombie_domains
  - machine
  - server_type
  - storage_channel
  - total_mem
  - total_zombie_mem
  - free_disk_space
  - free_mem
  - free_memory

25.3 Add Xen/KVM Compute Resource

To add a new Xen/KVM compute resource, use the following request:
POST /settings/hypervisors.xml
POST /settings/hypervisors.json

XML Request Example

```bash
curl -X POST http://onapp.test/settings/hypervisors.xml -d
  '<hypervisor><label>HV_LABEL</label><ip_address>HV_IP</ip_address><hypervisor_type>kvm/xen</hypervisor_type><segregation_os_type>any_os</segregation_os_type><enabled>true/false</enabled><disable_failover>true</disable_failover><failover_recipe_id>get_if_config</failover_recipe_id><collect_stats>1</collect_stats><hypervisor_group_id>HV_Group_id</hypervisor_group_id><backup_ip_address>192.168.123.1</backup_ip_address><cpu_units>1000</cpu_units></hypervisor>'
```

Where:

- **ip_address** - the compute resource IP address
- **backup_ip_address** - provisioning network IP address. Be aware, that it is not an IP address of a backup server, it is an IP address of an interface on a compute resource. It is used not to overload a management network.
- **label** - the name of the compute resource
- **hypervisor_type** - specify if this is Xen or KVM compute resource
- **segregation_os_type** - an operating system type (can be any_os, windows_only or non_windows)
- **enabled** - set true to enable a compute resource, otherwise set false
- **hypervisor_group_id** - the ID of the group to which this compute resource is assigned
- **collect_stats** - set 1 to collect statistics for this compute resource, otherwise set 0
- **disable_failover** - set true to disable compute resource failover, otherwise set false
- **failover_recipe_id** - the ID of a recipe to run before the failover process
- **cpu_units** - set the amount of CPU units for this compute resource

JSON Request Example

```bash
curl -X POST http://onapp.test/settings/hypervisors.json -d
  '{"hypervisor":{"label":"HV_LABEL","ip_address":"HV_IP","backup_ip_address":"192.168.123.1","hypervisor_type":"kvm/xen","segregation_os_type":"any_os","enabled":true,"disable_failover":true,"failover_recipe_id":"get_if_config","collect_stats":1,"hypervisor_group_id":"HV_Group_id","cpu_units":"1000"}}'
```

Page History

- v.5.9
  - added failover_recipe_id parameter
  - added segregation_os_type parameter
- v.3.3
• added cpu_units parameter

25.4 Add Static Compute Resource

To create static compute resource, use the following request:

POST /settings/hypervisors.xml
POST /settings/hypervisors.json

XML Request Example

curl -X POST http://onapp.test/settings/hypervisors.xml -d '  
<hypervisor><label>static</label><hypervisor_type>kvm</hypervisor_type><segregation_os_type>any_os</segregation_os_type><ip_address>191.168.1.148</ip_address><backup_ip_address>192.168.123.1</backup_ip_address><cpu_units>1000</cpu_units><enabled>1</enabled><collect_stats>1</collect_stats><disallow_failover>1</disallow_failover><failover_recipe_id>get_if_config</failover_recipe_id><amqp_exchange_name></amqp_exchange_name><static_integrated_storage>1</static_integrated_storage><mtu>1500</mtu><storage_bonding_mode>802.3ad</storage_bonding_mode><storage_controller_memory_size>1024</storage_controller_memory_size><storage_controller_db_size>128</storage_controller_db_size><disks_per_storage_controller>4</disks_per_storage_controller><storage_vlan>2</storage_vlan><power_cycle_command>#123</power_cycle_command><hypervisor>' -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml'

JSON Request Example

curl -X POST http://onapp.cloud/settings/hypervisors.json -d '{
  "hypervisor": {
    "label": "static",
    "hypervisor_type": "kvm",
    "segregation_os_type": "any_os",
    "ip_address": "191.168.1.148",
    "backup_ip_address": "192.168.123.1",
    "cpu_units": "1000",
    "enabled": "1",
    "collect_stats": "1",
    "disallow_failover": "1",
    "failover_recipe_id": "get_if_config",
    "amqp_exchange_name": "",
    "static_integrated_storage": "1",
    "mtu": "1500",
    "storage_bonding_mode": "802.3ad",
    "storage_controller_memory_size": "1024",
    "storage_controller_db_size": "128",
    "disks_per_storage_controller": "4",
    "storage_vlan": "2",
    "power_cycle_command": "#123"
  }
}' -u admin:password -H 'Accept: application/json' -H 'Content-Type: application/json'

Where:

hypervisor - an array of compute resource details:

• label " - the name of the compute resource
• hypervisor_type - compute resource type. For static compute resources, only KVM type can be used.
• segregation_os_type - an operating system type (can be Any OS, Windows only or Non-Windows)
• ip_address - the compute resource IP address
• backup_ip_address - provisioning network IP address
• cpu_units - set the amount of CPU units for this compute resource
- **enabled** - set 1 to enable this compute resource, otherwise set 0
- **collect_stats** - set 1 to collect statistics for this compute resource, otherwise set 0
- **disable_failover** - optional parameter. Set true to disable compute resource failover, otherwise false
- **failover_recipe_id** - the ID of a recipe to run before the failover process
- **static_integrated_storage** - set "1" to enable static integrated storage
- **mtu** - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.
- **storage_bonding_mode** - the type of bonding of the storage networks
- **storage_controller_memory_size** - specify the storage controller memory size (minimum 640 MB)
- **storage_controller_db_size** - specify the storage controller database size
- **disks_per_storage_controller** - specify the number of disks per controller virtual server. You can set form 1 to 4 disks. By default, the controller virtual server is created per 4 disk drives
- **storage_vlan** - the number of a VLAN this storage network belongs to
- **power_cycle_command** - arbitrary command string to be executed by IPMI from the CP server.

### 25.5 Add CloudBoot Compute Resource

To create a CloudBoot compute resource, use the following request:

POST /settings/assets/:asset_mac_address/hypervisors.xml

POST /settings/assets/:asset_mac_address/hypervisors.json

For details how to create CloudBoot compute resources for smart/baremetal server deployment, refer to the [Add Smart CloudBoot Compute Resource](#) and [Add Baremetal CloudBoot Compute Resource](#) sections, accordingly.

**XML Request Example**
curl -i -X POST
'(<hypervisor><label>CB_Virtual</label><pxe_ip_address_id>1248</pxe_ip_address_id><hypervisor_type>xen</hypervisor_type><segregation_os_type>any</segregation_os_type><server_type>virtual</server_type><backup>0</backup><backup_ip_address></backup_ip_address><collect_stats>1</collect_stats><disable_failover>0</disable_failover><format_disks>0</format_disks><passthrough_disks>0</passthrough_disks><storage><disks type="array"><disk><scsi>DC0710130DBA80013_TAII_DC0710130DBA80013</scsi><selected>1</selected></disk></disks><nics type="array"><nic><mac>00:30:48:fd:74:c7</mac><type>1</type></nic><nic><mac>00:1b:21:6f:3a:ff</mac><type>0</type></nic></nics><custom_pcis type="array"><custom_pci><pci>00:00.0</pci><selected>1</selected></custom_pci></custom_pcis><storage_controller_memory_size>640</storage_controller_memory_size><disks_per_storage_controller>4</disks_per_storage_controller><cloud_boot_os>centos5</cloud_boot_os><custom_config>iscsiadm -m discovery -t sendtargets -p 109.123.105.131\r\n/etc/init.d/iscsi restart</custom_config></hypervisor>' -u user:userpass
-H 'Accept:application/xml'
-H 'Content-type:application/xml'

Where:

hypervisor - an array of compute resource details:

- **label** - the name of the compute resource
- **pxe_ip_address_id** - the ID of Cloud Boot IP address that will be used for this compute resource
- **hypervisor_type** - compute resource type:
  - xen
  - kvm
- **segregation_os_type** - an operating system type (can be Any OS, Windows only or Non-Windows)
- **server_type** - specify the type of servers that will be deployed on this compute resource:

JSON Request Example

curl -i -X POST
-H 'Accept:application/json'
-H 'Content-type:application/json'
o virtual - specify the virtual server type to use this compute resource for virtual servers deployment. The server_type is virtual by default.

- backup - set 1 if you want to use this CloudBoot compute resource as a backup/transaction Server. OnApp Storage cannot be provisioned if this option is selected.
- backup_ip_address - provisioning network IP address
- enabled - set 1 to enable this compute resource, otherwise set 0
- collect_stats - set 1 to collect statistics for this compute resource, otherwise set 0
- disable_failover - optional parameter. Set true to disable compute resource failover, otherwise false
- failover_recipe_id - the ID of a recipe to run before the failover process
- format_disks - set 1 to format compute resource’s disks during creation, otherwise set 0
- passthrough_disks - set 1 if the disks should be passed through to the storage nodes, otherwise set 0. This parameter is for Xen compute resources only.

storage - an array of compute resource disks' details:
- disks - an array of compute resource disks, where:
  o scsi - SCSI inquiry product revision number
  o selected - set 1 to select a disk, otherwise set 0
- nics - network interfaces that will be used for storage, where:
  o mac - network interface MAC address
  o type - network interface type:
    - 0 - leave the NIC unused
    - 1 - SAN subnet - select this option to use this interface for storage network. In this case, NIC interface will be bonded with virtual network interface of the Storage Control Panel server
    - 2 - passthrough to storage - this option is available for Xen CloudBoot compute resources. The network interface will be added to the Storage Controller Server without the bond and the Storage Control Panel server will have the complete control over this interface
    - 3 - passthrough to Guest - this option is available for smart CloudBoot compute resources. The network interface will be added to the smart server
- custom_pcis - an array of custom PCI devices
  o pci - NIC PCI
  o selected - 1 if the PCI is selected, otherwise false
- mtu - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.

The maximum transportation unit (MTU) is the maximum size of a unit that can be transmitted transferred via ethernet traffic. Any data that exceed the specified MTU value will be divided into smaller units before being transferred. Utilization of jumbo frames allows to reduce increase throughput and increase CPU utilization during large size file
transfers.

- `storage_controller_memory_size` - specify the storage controller memory size (minimum 640 MB)

- `disks_per_storage_controller` - specify the number of disks per controller virtual server. You can set from 1 to 4 disks. By default, the controller virtual server is created per 4 disk drives

- `cloud_boot_os` - specify the compute resource OS. This parameter is for Xen compute resources only.

- `custom_config` - specify any custom commands you want to run when compute resource is booted

PLEASE NOTE: It is currently only possible to set NICs and disk information parameters via UI.

XML Output Example
<hypervisor>
  <allow_unsafe_assigned_interrupts type="boolean">true</allow_unsafe_assigned_interrupts>
  <backup type="boolean">false</backup>
  <cloud_boot_os>centos5</cloud_boot_os>
  <cpu_idle type="integer">0</cpu_idle>
  <cpu_mhz nil="true"/>
  <disks_per_storage_controller type="integer">4</disks_per_storage_controller>
  <disaster_recovery_recipe_id>get_if_config</disaster_recovery_recipe_id>
  <free_mem type="integer">0</free_mem>
  <ips nil="true"/>
  <list_of_logical_volumes nil="true"/>
  <list_of_volume_groups nil="true"/>
  <list_of_zombie_domains nil="true"/>
  <machines nil="true"/>
  <mem_info type="integer">0</mem_info>
  <mtu type="integer">1500</mtu>
  <numa_node type="integer">0</numa_node>
  <ovs nil="true"/>
  <passthrough_disks type="boolean">false</passthrough_disks>
  <release nil="true"/>
  <segregation_os_type>any</segregation_os_type>
  <server_type>virtual</server_type>
  <spared disk type="boolean">false</spared disk>
  <storage_channel>224.3.28.1</storage_channel>
  <storage_controller_memory_size type="integer">640</storage_controller_memory_size>
  <threads_per_core nil="true"/>
  <total_mem nil="true"/>
  <total_vmem nil="true"/>
  <updated_at type="datetime">2013-07-25T10:24:41+03:00</updated_at>
  <uptime nil="true"/>
  <vmware_total_cpu_cores type="integer">0</vmware_total_cpu_cores>
  <vms type="integer">0</vms>
</hypervisor>
<storage>
  <disks type="array">
    <disk>
      <name>sda</name>
      <scsi>DC0710130DBA80013_TAII_DC0710130DBA80013</scsi>
      <selected type="boolean">true</selected>
    </disk>
  </disks>
  <nics type="array">
    <nic>
      <name>eth1</name>
      <mac>00:30:48:fd:74:c7</mac>
      <type type="integer">1</type>
    </nic>
    <nic>
      <name>eth2</name>
      <mac>00:1b:21:6f:3a:ff</mac>
      <type type="integer">0</type>
    </nic>
  </nics>
  <custom_pcis type="array">
    <custom_pci>
      <name>Intel Corporation 5520/5500/X58 I/O Hub to ESI Port [8086:3405] (rev 13)</name>
      <pci>00:00.0</pci>
      <selected type="boolean">true</selected>
    </custom_pci>
  </custom_pcis>
</storage>
</hypervisor>

Page History

v.5.9
- added the following parameters:
  - segregation_os_type
  - failover_recipe_id

v3.1
- added the following parameters:
  - cloud_boot_os
  - storage_controller_memory_size
  - disks_per_storage_controller
  - custom_pcis
  - passthrough_custom_pcis
  - server_type
  - type (storage parameter)
25.6 Add Smart CloudBoot Compute Resource

To create a CloudBoot compute resource, use the following request:

POST /settings/assets/:asset_mac_address/hypervisors.xml

POST /settings/assets/:asset_mac_address/hypervisors.json

**XML Request Example**

```
curl -i -X POST http://onapp.test/settings/assets/00:30:48:fd:74:c6/hypervisors.xml -d "<hypervisor><label>smartzaza</label><pxe_ip_address_id>1248</pxe_ip_address_id><hypervisor_type>kvm</hypervisor_type><segregation_os_type>any</segregation_os_type><server_type>smart</server_type><backup_ip_address></backup_ip_address><hypervisor_type>kv
m</hypervisor_type><backup_ip_address></backup_ip_address><enabled>1</enabled><collect_stats>1</collect_stats><disable_failover>1</disable_failover><format_disks>1</format_disks><passthrough_disks>0</passthrough_disks><storage><disks type="array"><scsi>DC0710130DBA80013_TAI1_DC0710130DBA80013</scsi><selected>1</selected></disk></storage><nics type="array"><nic><mac>00:25:90:10:62:5b</mac><type>1</type></nic><nic><mac>00:1b:21:72:9d:06</mac><type>3</type></nic></nics><storage_controller_memory_size>640</storage_controller_memory_size><disks_per_storage_controller>4</disks_per_storage_controller><allow_unsafe_assigned_interrupts>true</allow_unsafe_assigned_interrupts><custom_config></custom_config></hypervisor>"
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X POST http://onapp.test/settings/assets/00:30:48:fd:74:c6/hypervisors.json -d '{"hypervisor":{"label":"smart","pxe_ip_address_id":"2","hypervisor_type":"kvm","segregation_os_type":"any","server_type":"smart","backup_ip_address":"
,"enabled":"1","collect_stats":"1","disable_failover":"1","format_disks":"1"",
,"passthrough_disks":"0"},"storage":{"disks":[{"scsi":"9VM51JELS_9VM51JEL","selected":"1"}]
},"nics":[{"mac":"00:25:90:10:62:5b","type":"1"}],
{"mac":"00:1b:21:72:9d:06","type":"3"}],
"custom_pcs":[{"pci":"00:00.0","selected":"1"}],
"passthrough_custom_pcs":1,"mtu":1500,"storage_controller_memory_size":640,"disks_per_storage_controller":4,
"allow_unsafe_assigned_interrupts":true,"custom_config":null"}' -u user:userpass -H 'Accept:application/json' -H 'Content-type: application/json'
```

Where:

- **hypervisor** - an array of compute resource details:
  - *label* - the name of the compute resource
  - *pxe_ip_address_id* - the ID of Cloud Boot IP address that will be used for this compute resource
  - *hypervisor_type* - compute resource type:
You can only specify the KVM type for smart CloudBoot compute resource provisioning.

- `segregation_os_type` - an operating system type (can be Any OS, Windows only or Non-Windows)

- `server_type` - specify the type of servers that will be deployed on this compute resource:
  - `smart` - specify the smart server type to use this compute resource for smart server deployment. The `server_type` is virtual by default.

- `backup_ip_address` - provisioning network IP address

- `enabled` - set 1 to enable this compute resource, otherwise set 0

- `collect_stats` - set 1 to collect statistics for this compute resource, otherwise set 0

- `disable_failover` - optional parameter. Set true to disable compute resource failover, otherwise false

- `failover_recipe_id` - the ID of a recipe to run before the failover process

- `format_disks` - set 1 to format compute resource's disks during creation, otherwise set 0

- `passthrough_disks` - set 1 if the disks should be passed through to the storage nodes, otherwise set 0. This parameter is for Xen compute resources only.

`storage` - an array of compute resource disks' details:

- `disks` - an array of compute resource disks, where:
  - `scsi` - SCSI inquiry product revision number
  - `selected` - set 1 to select a disk, otherwise set 0

- `nics` - network interfaces that will be used for storage, where:
  - `mac` - network interface MAC address
  - `type` - network interface type:
    - 0 - leave the NIC unused
    - 1 - SAN subnet - select this option to use this interface for storage network. In this case, NIC interface will be bonded with virtual network interface of the Storage Controller Server
    - 2 - passthrough to storage - this option is available for Xen CloudBoot compute resources. The network interface will be added to the Storage Controller Server without the bond and the Storage Controller Server will have the complete control over this interface
    - 3 - passthrough to Guest - this option is available for smart CloudBoot compute resources. The network interface will be added to the smart server

- `custom_pcis` - an array of custom PCI devices
  - `pci` - NIC PCI
- **selected** - 1 if the PCI is selected, otherwise false

- **mtu** - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.

The maximum transportation unit (MTU) is the maximum size of a unit that can be transmitted via ethernet traffic. Any data that exceed the specified MTU value will be divided into smaller units before being transferred. Utilization of jumbo frames allows to reduce increase throughput and increase CPU utilization during large size file transfers.

- **storage_controller_memory_size** - specify the storage controller memory size (minimum 640 MB)

- **disks_per_storage_controller** - specify the number of disks per controller virtual server. You can specify from 1 to 4 disks. By default, the controller virtual server is created per 4 disk drives

  *allow_unsafe_assigned_interrupts* - set true if the compute resource does not support the interrupt remapping, otherwise set false. Note: With *allow_unsafe_assigned_interrupts* parameter enabled, smart server is exposed to the PCI passthrough MSI trap injection.

  *custom_config* - specify any custom commands you want to run when compute resource is booted

  **PLEASE NOTE:** It is currently only possible to set NICs and disk information parameters via UI.

**XML Output Example**
<hypervisor>
  <allow_unsafe_assigned_interrupts type="boolean">true</allow_unsafe_assigned_interrupts>
  <backup type="boolean">false</backup>
  <backup_ip_address nil="true"/>
  <block type="boolean">true</block>
  <built type="boolean">false</built>
  <called_in_at nil="true"/>
  <cloud_boot_os>centos5</cloud_boot_os>
  <connection_options nil="true"/>
  <cpu_idle type="integer">0</cpu_idle>
  <cpu_mhz nil="true"/>
  <cpus nil="true"/>
  <created_at type="datetime">2013-07-25T10:24:41+03:00</created_at>
  <custom_config>iscsiadm -m discovery -t sendtargets -p 109.123.105.131\r\n/etc/init.d/iscsi restart</custom_config>
  <disable_failover type="boolean">false</disable_failover>
  <disks_per_storage_controller type="integer">4</disks_per_storage_controller>
  <distro nil="true"/>
  <enabled type="boolean">true</enabled>
  <failover_recipe_id>get_if_config</failover_recipe_id>
  <failure_count type="integer">0</failure_count>
  <format_disks type="boolean">false</format_disks>
  <free_mem type="integer">0</free_mem>
  <host nil="true"/>
  <host_id type="integer">2</host_id>
  <hypervisor_group_id nil="true"/>
  <hypervisor_type>xen</hypervisor_type>
  <id type="integer">43</id>
  <ip_address>109.123.105.133</ip_address>
  <label>CB_Virtual</label>
  <list_of_logical_volumes nil="true"/>
  <list_of_volume_groups nil="true"/>
  <list_of_zombie_domains nil="true"/>
  <locked type="boolean">false</locked>
  <mac>00:30:48:fd:74:c6</mac>
  <machine nil="true"/>
  <mem_info type="integer">0</mem_info>
  <mtu type="integer">1500</mtu>
  <online type="boolean">false</online>
  <ovs nil="true"/>
  <passthrough_disks type="boolean">false</passthrough_disks>
  <release nil="true"/>
  <segregation_os_type>any</segregation_os_type>
  <server_type>virtual</server_type>
  <spare type="boolean">false</spare>
  <storage_channel>224.3.28.1</storage_channel>
  <storage_controller_memory_size type="integer">640</storage_controller_memory_size>
  <threads_per_core nil="true"/>
  <total_mem nil="true"/>
  <total_zombie_mem nil="true"/>
  <updated_at type="datetime">2013-07-25T10:24:41+03:00</updated_at>
  <uptime nil="true"/>
  <vmware_total_cpu_cores type="integer">0</vmware_total_cpu_cores>
  <used_cpu_resources type="integer">0</used_cpu_resources>
  <total_cpu type="integer">0</total_cpu>
  <total_memory type="integer">0</total_memory>
  <cpu_cores type="integer">0</cpu_cores>
  <free_disk_space nil="true"/>
  <memory_allocated_by_running_vms type="integer">0</memory_allocated_by_running_vms>
  <total_memory_allocated_by_vms type="integer">0</total_memory_allocated_by_vms>
</hypervisor>
<storage>
  <disks type="array">
    <disk>
      <name>sda</name>
      <scsi>DC0710130DBA80013_TAII_DC0710130DBA80013</scsi>
      <selected type="boolean">true</selected>
    </disk>
  </disks>
  <nics type="array">
    <nic>
      <name>eth1</name>
      <mac>00:30:48:fd:74:c7</mac>
      <type type="integer">1</type>
    </nic>
    <nic>
      <name>eth2</name>
      <mac>00:1b:21:6f:3a:ff</mac>
      <type type="integer">0</type>
    </nic>
  </nics>
  <custom_pcis type="array">
    <custom_pci>
      <name>Intel Corporation 5520/5500/X58 I/O Hub to ESI Port [8086:3405] (rev 13)</name>
      <pci>00:00.0</pci>
      <selected type="boolean">true</selected>
    </custom_pci>
  </custom_pcis>
</storage>

**Page History**

v.5.9
- added the following parameters:
  - segregation_os_type
  - failover_recipe_id

v3.1
- added the following parameters:
  - allow_unsafe_assigned_interrupts
  - cloud_boot_os
  - storage_controller_memory_size
  - disks_per_storage_controller
  - custom_pcis
  - passthrough_custom_pcis
  - server_type
  - type (storage parameter)
25.7 Add Baremetal CloudBoot Compute Resource

To create a CloudBoot compute resource, use the following request:

POST /settings/assets/:asset_mac_address/hypervisors.xml

POST /settings/assets/:asset_mac_address/hypervisors.json

**XML Request Example**

```
curl -i -X POST
  '<hypervisor><label>baremetal</label><pxe_ip_address_id>2</pxe_ip_address_id>
  <hypervisor_type>kvm</hypervisor_type><server_type>baremetal</server_type>
  <enabled>1</enabled><failover_recipe_id>get_if_config</failover_recipe_id></hypervisor>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X POST
  '{"hypervisor":{"label":"baremetal","pxe_ip_address_id":2,"hypervisor_type":"kvm","server_type":"baremetal","enabled":1,"failover_recipe_id":"get_if_config"}}' -u user:userpass -H 'Accept:application/json' -H 'Content-type: application/json'
```

Where:

- **hypervisor** - an array of compute resource details:
  - *label* - the name of the compute resource
  - *pxe_ip_address_id* - the ID of Cloud Boot IP address that will be used for this compute resource
  - **hypervisor_type** - specify if this is Xen or KVM compute resource
  - **server_type** - specify the type of servers that will be deployed on this compute resource:
    - *baremetal* - specify the baremetal server type to use this compute resource for baremetal server deployment. The **server_type** is virtual by default.
  - *enabled* - set 1 to enable this compute resource, otherwise set 0
  - **failover_recipe_id** - the ID of a recipe to run before the failover process

**Page History**

- **v.6.1**
  - added KVM as an option to the **hypervisor_type** parameter
- **v.5.9**
  - added the **failover_recipe_id** parameter
- **v. 3.1 RC1**
  - added the following parameters:
    - *server_type*
25.8 Add VMware Compute Resource

To add a VMware compute resource, use the following request:

POST /settings/hypervisors.xml
POST /settings/hypervisors.json

**XML Request Example**

```
curl -i -X POST http://onapp.test/settings/hypervisors.xml -d '  
  <hypervisor><label>zaza_fake_ware_xml</label><ip_address>10.10.10.12</ip_address><backup_ip_address>10.10.10.13</backup_ip_address><hypervisor_type>vmware</hypervisor_type><segregation_os_type>any</segregation_os_type><enabled>1</enabled><collect_stats>1</collect_stats><disable_failover>1</disable_failover><connection_options><login>login</login><password>password</password><cluster_name>OnApp</cluster_name><distributed_virtual_switch_name>dvSwitch</distributed_virtual_switch_name></connection_options></hypervisor>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X POST http://onapp.test/settings/hypervisors.json -d '{
  "hypervisor": {
    "label": "zaza_fake_ware_json",
    "ip_address": "10.10.10.2",
    "backup_ip_address": "10.10.10.3",
    "hypervisor_type": "vmware",
    "segregation_os_type": "any",
    "enabled": "1",
    "collect_stats": "1",
    "disable_failover": "1",
    "connection_options": {
      "login": "login",
      "password": "password",
      "cluster_name": "OnApp",
      "distributed_virtual_switch_name": "dvSwitch"
    }
  }
}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **ip_address** - the compute resource IP address
- **backup_ip_address** - provisioning network IP address. Be aware, that it is not an IP address of a backup server, it is an IP address of an interface on a compute resource. It is used not to overload a management network.
- **label** - the name of the compute resource
- **hypervisor_type** - specify if this is Xen or KVM compute resource
- **segregation_os_type** - an operating system type (can be Any OS, Windows only or Non-Windows)
- **enabled** - optional parameter, set True to enable a compute resource
- **hypervisor_group_id** - the ID of the group to which this compute resource is assigned
- **collect_stats** - set 1 to collect statistics for this compute resource, otherwise set 0
- **disable_failover** - optional parameter. Set true to disable compute resource failover, otherwise false
- **connection_options** - an array of the following vCenter cluster parameters:
  - **login** - vCenter login
• password - vCenter password
• cluster_name - vCenter cluster name
• distributed_virtual_switch_name - distributed virtual switch label

Page History
v.5.9
• added the segregation_os_type parameter

25.9 Edit Xen/KVM Compute Resource

To edit Xen/KVM compute resource, use the following request:

PUT /settings/hypervisors/:id.xml
PUT /settings/hypervisors/:id.json

XML Request Example

```bash
curl -X PUT http://onapp.test/settings/hypervisors/13.xml -d '<hypervisor>
    <label>HV_LABEL</label>
    <ip_address>HV_IP</ip_address>
    <hypervisor_type>kvm/xen</hypervisor_type>
    <enabled>true/false</enabled>
    <disable_failover>true/false</disable_failover>
    <failover_recipe_id>get_if_config</failover_recipe_id>
    <backup_ip_address>192.168.123.1</backup_ip_address>
    <segregation_os_type>any_os</segregation_os_type>
    <hypervisor_group_id>HV_Group_id</hypervisor_group_id>
    <cpu_units>1000</cpu_units>
</hypervisor>' -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml'
```

JSON Request Example

```bash
curl -X PUT http://onapp.test/settings/hypervisors/13.json -d '{
    "hypervisor": {
        "label": "HV_LABEL",
        "ip_address": "HV_IP",
        "hypervisor_type": "kvm/xen",
        "enabled": "true/false",
        "disable_failover": "true/false",
        "failover_recipe_id": "get_if_config",
        "backup_ip_address": "192.168.123.1",
        "segregation_os_type": "any_os",
        "hypervisor_group_id": "HV_Group_id"
    }
}' -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json'
```

Where:

- **ip_address** - the compute resource IP address
- **label** - the name of the compute resource

**backup_ip_address** - provisioning network IP address. Be aware, that it is not an IP address of a backup server, it is an IP address of an interface on a compute resource. It is used not to overload a management network.

- **hypervisor_type** - compute resource type
- **segregation_os_type** - an operating system type (can be Any OS, Windows only or Non-Windows)

- **enabled** - set true to enable a compute resource, otherwise set false

- **hypervisor_group_id** - set ID of the compute zone to attach this compute resource to it, or send the empty value to remove the compute resource from the compute zone

- **disable_failover** – set true to disable compute resource failover, otherwise set false
failover_recipe_id - the ID of a recipe to run before the failover process

cpu_units - set the amount of CPU units for this compute resource

Returns HTTP 204 response on successful deletion, or HTTP 404 when a compute resource with the ID specified is not found, or the URL requested is incorrect.

Page History

v.5.9
- added the following parameters:
  - segregation_os_type
  - failover_recipe_id

v. 3.3
- added cpu_units parameter

25.10 Edit Static Compute Resource

To edit static compute resource, use the following request:

PUT /settings/hypervisors/:hypervisor_id.xml

PUT /settings/hypervisors/:hypervisor_id.json

XML Request Example

curl -X PUT http://onapp.test/settings/hypervisors/25.xml -d
  '<hypervisor><label>static</label><hypervisor_type>kvm</hypervisor_type><segregation_os_type>any_os</segregation_os_type><ip_address>191.168.1.148</ip_address><backup_ip_address>192.168.123.148</backup_ip_address><cpu_units>1000</cpu_units><enabled>1</enabled><collect_stats>1</collect_stats><disable_failover>1</disable_failover><failover_recipe_id>get_if_config</failover_recipe_id><amqp_exchange_name></amqp_exchange_name><static_integrated_storage>1</static_integrated_storage><mtu>1500</mtu><storage_bonding_mode>802.3ad</storage_bonding_mode><storage_controller_memory_size>1024</storage_controller_memory_size><storage_controller_db_size>128</storage_controller_db_size><disks_per_storage_controller>4</disks_per_storage_controller><storage_vlan>2</storage_vlan><power_cycle_command>#222123</power_cycle_command></hypervisor>' -u user:password -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

Where:

- **label** - the name of the compute resource
- **hypervisor_type** - compute resource type. For static compute resources, only KVM type can be used.
- **segregation_os_type** - an operating system type (can be Any OS, Windows only or Non-Windows)
- **ip_address** - the compute resource IP address
- **backup_ip_address** - provisioning network IP address
- **cpu_units** - set the amount of CPU units for this compute resource
- **enabled** - set 1 to enable this compute resource, otherwise set 0
- **collect_stats** - set 1 to collect statistics for this compute resource, otherwise set 0
- **disable_failover** - optional parameter. Set true to disable compute resource failover, otherwise false
- **failover_recipe_id** - the ID of a recipe to run before the failover process
- **static_integrated_storage** - set "1" to enable static integrated storage
- **mtu** - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.
- **storage_bonding_mode** - the type of bonding of the storage networks
- **storage_controller_memory_size** - specify the storage controller memory size (minimum 640 MB)
- **storage_controller_db_size** - specify the storage controller database size
- **disks_per_storage_controller** - specify the number of disks per controller virtual server. You can set form 1 to 4 disks. By default, the controller virtual server is created per 4 disk drives
- **storage_vlan** - the number of a VLAN this storage network belongs to
- **power_cycle_command** - arbitrary command string to be executed by IPMI from the CP server

### 25.11 Edit CloudBoot Compute Resource

To edit a virtual CloudBoot compute resource, use the following request:

PUT /settings/assets/:asset_mac_address/hypervisors.xml

PUT /settings/assets/:asset_mac_address/hypervisors.json
For details how to edit CloudBoot compute resources for smart/baremetal server deployment, refer to the Edit Smart CloudBoot Compute Resource and Edit Baremetal CloudBoot Compute Resource sections, accordingly.

**XML Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/assets/00:0a:95:9d:68:16/hypervisors.xml -d '<hypervisor><label>CB_Virtual</label><ip_address>10.0.52.2</ip_address><backup_ip_address><segregation_os_type>any</segregation_os_type><enabled>1</enabled><collect_stats>1</collect_stats><disable_failover>0</disable_failover><failover_recipe_id>get_if_config</failover_recipe_id><passsthrough_disks>0</passsthrough_disks><storage><disks><scsi>DC0710130DBA80013_TAII_DC0710130DBA80013</scsi><selected>1</selected></disks><nics><mac>00:30:48:fd:74:c7</mac><type>1</type></nics><nics><mac>00:1b:21:6f:3a:ff</mac><type>0</type></nics><custom_pcis><pci>00:00.0</pci><selected>1</selected></custom_pcis></storage><mtu>1500</mtu><storage_controller_memory_size>640</storage_controller_memory_size><disks_per_storage_controller>4</disks_per_storage_controller><integrated_storage_disabled>false</integrated_storage_disabled><custom_config>iscsiadm -m discovery -t sendtargets -p 109.123.105.131 \n\r\n\r\n/etc/init.d/iscsi restart</custom_config><apply_hypervisor_group_custom_config>0</apply_hypervisor_group_custom_config></hypervisor>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/assets/00:0a:95:9d:68:16/hypervisors.json -d '{"hypervisor":{"label":"virtual","ip_address":"10.0.52.2","backup_ip_address":null,"segregation_os_type":"any","enabled":"1","collect_stats":1,"disable_failover":0,"failover_recipe_id":"get_if_config","passsthrough_disks":0,"storage":{"disks": [{"scsi":"DC0710130DBA80013_TAII_DC0710130DBA80013","selected":1}],"nics": [{"mac":"00:30:48:fd:74:c7","type":1}, {"mac":"00:1b:21:6f:3a:ff","type":0}],"custom_pcis": [{"pci":"00:00.0","selected":1}],"mtu":1500,"storage_controller_memory_size":640,"disks_per_storage_controller":4,"integrated_storage_disabled":false,"custom_config":"iscsiadm -m discovery -t sendtargets -p 109.123.105.131 \n\r\n\r\n/etc/init.d/iscsi restart","apply_hypervisor_group_custom_config":1}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **hypervisor** - an array of compute resource details:
  - **label** - the name of the compute resource
  - **ip_address** - the compute resource IP address
  - **ip_backup_address** - the compute resource IP address
  - **segregation_os_type** - the type of the compute resource
  - **enabled** - whether the compute resource is enabled
  - **collect_stats** - whether the compute resource collects statistics
  - **disable_failover** - whether the compute resource is disabled for failover
  - **failover_recipe_id** - the failover recipe ID
  - **passsthrough_disks** - whether the compute resource passes through disks
  - **storage** - details of the storage
    - **disks** - an array of disk details
      - **scsi** - the disk SCSI ID
      - **selected** - whether the disk is selected
    - **nics** - an array of network interface card (NIC) details
      - **mac** - the NIC MAC address
      - **type** - the NIC type
    - **custom_pcis** - an array of custom PCI details
      - **pci** - the PCI address
      - **selected** - whether the PCI is selected
  - **mtu** - the MTU size
  - **storage_controller_memory_size** - the storage controller memory size
  - **disks_per_storage_controller** - the number of disks per storage controller
  - **integrated_storage_disabled** - whether the integrated storage is disabled
  - **custom_config** - custom configuration commands
  - **apply_hypervisor_group_custom_config** - whether to apply the custom configuration to the hypervisor group

When you change the Pxe IP address, you should reboot the CloudBoot compute resource immediately after saving the new settings. If you do not reboot the resource immediately, the Control Panel will fail to connect to the new IP address, causing failover transactions. You can reboot the compute resource manually from
the console or use the Power Cycle command (if configured). You cannot use the Reboot option on the CP UI to reboot the resource after changing the IP address. You can also change the IP address of a CloudBoot compute resource that is offline and once the resource is booted, it will be available on the new IP Address.

- If InfiniBand is enabled for CloudBoot, you should change a value of the cloud_boot pxe config after changing the Pxe IP address.

- backup_ip_address - the provisioning network IP address
- segregation_os_type - an operating system type (can be Any OS, Windows only or Non-Windows)
- enabled - set 1 to enable this compute resource, otherwise, set 0
- collect_stats - set 1 to collect statistics for this compute resource, otherwise, set 0
- disable_failover - optional parameter. Set true to disable compute resource failover, otherwise, false
- failover_recipe_id - the ID of a recipe to run before the failover process
- format_disks - set 1 to format compute resource's disks during creation, otherwise, set 0
- passthrough_disks - set 1 if the disks should be passed through to the storage nodes, otherwise set 0. This parameter is for Xen compute resources only.

storage - an array of compute resource disks' details:
- disks - an array of compute resource disks, where:
  - scsi - SCSI inquiry product revision number
  - selected - set 1 to select a disk, otherwise set 0
- nics - network interfaces that will be used for storage, where:
  - mac - network interface MAC address
  - type - network interface type:
    - 0 - unassigned (is not used for storage)
    - 1 - SAN subnet
    - 2 - passthrough to storage
    - 3 - passthrough to guest (for smart appliance)
- custom_pcis - an array of custom PCI devices
  - pci - NIC PCI
  - selected - 1 if the PCI is selected, otherwise false
- mtu - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.

The maximum transportation unit (MTU) is the maximum size of a unit that can be transmitted transferred via ethernet traffic. Any data that exceed the specified MTU value will be divided into smaller units.
before being transferred. Utilization of jumbo frames allows to reduce increase throughput and increase CPU utilization during large size file transfers.

- **storage_controller_memory_size** - specify the storage controller memory size (minimum 640 MB)
- **disks_per_storage_controller** - specify the number of disks per controller virtual server. You can set from 1 to 4 disks. By default, the controller virtual server is created per 4 disk drives
- **power_cycle_command** - arbitrary command string to be executed by IPMI from the CP server
- **integrated_storage_disabled** - set 'true' to disable Integrated Storage
- **custom_config** - specify any custom commands that will be run when a compute resource is booted

Currently, a command or commands should be written in one line separated by a semicolon. If the command(s) is written in two lines you will receive a "fail" response, although the transaction will be performed. The power cycle command is executed on Control Panel under user OnApp, this may be any script created in bash.

**apply_hypervisor_group_custom_config** - optional parameter. Set 1 to enable applying the compute zone custom config, otherwise, set 0

If enabled, the compute zone custom config is applied before the compute resource custom config.

**Page History**

v.5.9
- added the following parameters:
  - **segregation_os_type**
  - **failover_recipe_id**

v.5.7
- added the **ip_address** parameter

v.5.6
- added the **apply_hypervisor_group_custom_config** parameter

v.5.4
- added the **integrated_storage_disabled** parameter

### 25.12 Edit Smart CloudBoot Compute Resource

To edit a smart CloudBoot compute resource, use the following request:
PUT /settings/assets/:asset_mac_address/hypervisors.xml
PUT /settings/assets/:asset_mac_address/hypervisors.json

**XML Request Example**

curl -i -X PUT
http://onapp.test/settings/assets/00:0a:95:9d:68:16/hypervisors.xml -d '
<hypervisor><label>smart</label><ip_address>10.0.52.2</ip_address><backup_ip_address></backup_ip_address><segregationToken>any</segregationToken><enabled>1</enabled><collect_stats>0</collect_stats><storeSize>0</storeSize><disableFailover>0</disableFailover><failoverRecipeId>get_if_config</failoverRecipeId>'
<storage><disks type="array"><disk><scsi>DC0710130DBA80013_TAII_DC0710130DBA80013</scsi><selected>1</selected></disk></disks><nics><mac>00:30:48:fd:74:c7</mac><type>1</type></nics><nics><mac>00:1b:21:5f:3a:ff</mac><type>3</type></nics></storage><mtu>1500</mtu><storageControllerMemorySize>640</storageControllerMemorySize><disksPerStorageController>4</disksPerStorageController><allowUnsafeAssignedInterrupts>0</allowUnsafeAssignedInterrupts><customConfig></customConfig></hypervisor>'

**JSON Request Example**

curl -i -X PUT

**Where:**

- **hypervisor** - an array of compute resource details:
  
  - **label** - the name of the compute resource
  - **ip_address** - the compute resource IP address

  - When you change the PXE IP address, you should reboot the CloudBoot compute resource immediately after saving the new settings. If you do not reboot the resource immediately, the Control Panel will fail to connect to the new IP address, causing failover transactions. You can reboot the compute resource manually from the console or use the Power Cycle command (if configured). You cannot use the Reboot option on the CP UI to reboot the resource after changing the IP address.
  You can also change the IP address of a CloudBoot compute resource that is offline and once the resource is booted, it will be available on the new IP Address.
- If InfiniBand is enabled for CloudBoot, you should change a value of the `cloud_boot` pxe config after changing the Pxe IP address.

- `backup_ip_address` - provisioning network IP address
- `segregation_os_type` - an operating system type (can be Any OS, Windows only or Non-Windows)

- `enabled` - set 1 to enable this compute resource, otherwise set 0
- `collect_stats` - set 1 to collect statistics for this compute resource, otherwise set 0
- `disable_failover` - optional parameter. Set true to disable compute resource failover, otherwise false
- `failover_recipe_id` - the ID of a recipe to run before the failover process

- `format_disks` - set 1 set 1 to format compute resource's disks during creation, otherwise set 0
- `passthrough_disks` - set 1 if the disks should be passed through to the storage nodes, otherwise set 0. This parameter is for Xen compute resources only.

`storage` - an array of compute resource disks' details:

- `disks` - an array of compute resource disks, where:
  - `scsi` - SCSI inquiry product revision number
  - `selected` - set 1 to select a disk, otherwise set 0
- `mtu` - maximum transferrable unit value. You can set the frame size from 1500 to 9000 bytes.

The maximum transportation unit (MTU) is the maximum size of a unit that can be transmitted transferred via ethernet traffic. Any data that exceed the specified MTU value will be divided into smaller units before being transferred. Utilization of jumbo frames allows to reduce increase throughput and increase CPU utilization during large size file transfers.

- `storage_controller_memory_size` - specify the storage controller memory size (minimum 640 MB)
- `disks_per_storage_controller` - specify the number of disks per controller virtual server. You can specify from 1 to 4 disks. By default, the controller virtual server is created per 4 disk drives

- `allow_unsafe_assigned_interrupts` - set true if the compute resource does not support the interrupt remapping, otherwise set false. Note: With allow_unsafe_assigned_interrupts parameter enabled, smart server is exposed to the PCI passthrough MSI trap injection.

- `custom_config` - specify any custom commands you want to run when compute resource is booted
PLEASE NOTE: It is currently only possible to set NICs and disk information parameters via UI.

Page History
v.5.9
- added the following parameters:
  o segregation_os_type
  o failover_recipe_id

v.5.7:
- added the ip_address parameter

25.13 Edit Baremetal CloudBoot Compute Resource

To edit a baremetal CloudBoot compute resource, use the following request:

```
PUT /settings/assets/:asset_mac_address/hypervisors.xml
PUT /settings/assets/:asset_mac_address/hypervisors.json
```

XML Request Example
```
curl -i -X PUT http://onapp.test/settings/assets/00:0a:95:9d:68:16/hypervisors.xml -d 'hypervisor'"label":"baremetal","ip_address":"10.0.52.2","enabled":"1"'"failover_recipe_id":"get_if_config"' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example
```
curl -i -X PUT http://onapp.test/settings/assets/00:0a:95:9d:68:16/hypervisors.json -d '{"hypervisor":{"label":"baremetal","ip_address":"10.0.52.2","enabled":"1","failover_recipe_id":"get_if_config"}}' -u user:userpass -H 'Accept:application/json' -H 'Content-type: application/json'
```

Where:
- **label** - the name of the compute resource
- **ip_address** - the compute resource IP address

- When you change the Pxe IP address, you should reboot the CloudBoot compute resource immediately after saving the new settings. If you do not reboot the resource immediately, the Control
Panel will fail to connect to the new IP address, causing failover transactions. You can reboot the compute resource manually from the console or use the Power Cycle command (if configured).
You cannot use the Reboot option on the CP UI to reboot the resource after changing the IP address.
You can also change the IP address of a CloudBoot compute resource that is offline and once the resource is booted, it will be available on the new IP Address.
- If InfiniBand is enabled for CloudBoot, you should change a value of the cloud_boot pxe config after changing the Pxe IP address.

- enabled - set 1 to enable this compute resource, otherwise set 0
- failover_recipe_id - the ID of a recipe to run before the failover process

Page History
v.5.9
- added the failover_recipe_id parameter
v.5.7
- added the ip_address parameter

25.14 Edit VMware Compute Resource

To edit a VMware compute resource, use the following request:
PUT /settings/hypervisors/:hypervisor_id.xml
PUT /settings/hypervisors/:hypervisor_id.json

XML Request Example
```
curl -i -X PUT http://onapp.test/settings/hypervisors/13.xml -d
  '<hypervisor><label>zaza_fake_ware.xml_ch</label><ip_address>10.10.10.12</ip_address><backup_ip_address>10.10.10.13</backup_ip_address><segregation_os_type>any</segregation_os_type><enabled>1</enabled><collect_stats>1</collect_stats><disable_failover>1</disable_failover><connection_options><login>login</login><password>password</password><cluster_name>OnApp</cluster_name><distributed_virtual_switch_name>dvSwitch</distributed_virtual_switch_name></connection_options></hypervisor>'
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example
curl -i -X PUT http://onapp.test/settings/hypervisors/13.json -d '{"hypervisor":{"label":"zaza_fake_ware_json","ip_address":"10.10.10.2","backup_ip_address":"10.10.10.3","segregation_os_type":"any","enabled":"1","collect_stats":"
"backup","disable_failover":"
"connection_options":{"login":"login","password":"password","cluster_name":"OnApp","distributed_virtual_switch_name":"dvSwitch"}}}'
-u user:userpass
-H 'Accept: application/json' -H 'Content-type:application/json'

Where:

- **ip_address** - the compute resource IP address
- **backup_ip_address** - provisioning network IP address. Be aware, that it is not an IP address of a backup server, it is an IP address of an interface on a compute resource. It is used not to overload a management network.
- **label** - the name of the compute resource
- **segregation_os_type** - an operating system type (can be Any OS, Windows only or Non-Windows)
- **enabled** - optional parameter, set True to enable a compute resource
- **collect_stats** - set 1 to collect statistics for this compute resource, otherwise set 0
- **disable_failover** - optional parameter. Set true to disable compute resource failover, otherwise false
- **connection_options** - an array of the following vCenter cluster parameters:
  - **login** - vCenter login
  - **password** - vCenter password
  - **cluster_name** - vCenter cluster name
  - **distributed_virtual_switch_name** - distributed virtual switch label

Page History

v.5.9

- removed the **hypervisor_type** parameter
- added the **segregation_os_type** parameter

### 25.15 Reboot Compute Resource

To reboot the compute resource, use the following request:

PUT /settings/hypervisors/:hypervisor_id/reboot.xml
PUT /settings/hypervisors/:hypervisor_id/reboot.json

XML Request Example

**JSON Request Example**

```
```

Where:

- **skip_powered_off_vms_migration** - set 1 to prevent the migration of powered off virtual servers to another compute resource during the reboot
- **schedule_failover** - set 1 to start running virtual servers after the reboot
- **force** - set 1 to stop all virtual servers that cannot be migrated to another compute resource
- **confirm** - set 1 to confirm the reboot of this compute resource

An HTTP 201 response is returned on a successful reboot. Unsuccessful reboot responses include HTTP 404 (resource not found – e.g. if the compute resource isn't online) and HTTP 422 (request cannot be processed – e.g. if parameters were incorrect).

**PLEASE NOTE:** Reboot option is not available for VMware compute resources.

**Page History**

- v. 6.0

- added the **skip_powered_off_vms_migration** parameter

**25.16 Delete Compute Resource**

To delete a compute resource, use the following request:

DELETE /settings/hypervisors/:id.xml
DELETE /settings/hypervisors/:id.json

**XML Request Example**

```
```

**JSON Request Example**
OnApp Cloud 6.3 API Guide

312

```
```

Returns HTTP 204 response on successful deletion, or HTTP 404 when a compute resource with the ID specified is not found, or the URL requested is incorrect.

### 25.17 Get List of Appliances Running on Compute Resource

To get the list of appliances running on a compute resource, use the following request:

GET /hypervisors/:hypervisor_id/virtual_machines.xml
GET /hypervisors/:hypervisor_id/virtual_machines.json

This request returns the list of all appliances deployed on the compute resource. For more information about particular appliances, see:

- **Get List of VSS** section for virtual servers
- **Get List of ASs** section for application servers
- **Get List of Smart Servers** section for smart servers
- **Get List of Baremetal Servers** section for baremetal servers
- **Get List of Load Balancers** section for load balancers
- **Get List of Accelerators** section for CDN accelerators
- **Get List of CDN Edge Servers** section for CDN edge servers

### 25.18 Get List of Data Store Joins Attached to Compute Resource

To get the list of compute resource data store joins (data stores which are attached to the compute resource), use the following request:

GET /settings/hypervisors/:hypervisor_id/data_store_joins.xml
GET /settings/hypervisors/:hypervisor_id/data_store_joins.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**
Where:

data_store_id - the ID of the data store, which is attached to the compute resource
hypervisor_id - the compute resource ID
id - the join ID
target_join_id - the ID of the join target; in this case it is the compute resource ID
target_join_type - type of join target; in this case it is compute resource

25.19 Get List of Data Stores Attached to Compute Resource

To get the list of data stores attached to the compute resource, use the following request:

GET /settings/hypervisor_zones/:hypervisor_zone_id/data_stores.xml
GET /settings/hypervisor_zones/:hypervisor_zone_id/data_stores.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example
<data_stores type="array">
  <data_store>
    <created_at type="datetime">2012-04-03T16:07:00+00:00</created_at>
    <data_store_group_id type="integer">3</data_store_group_id>
    <data_store_size type="integer">460</data_store_size>
    <enabled type="boolean">true</enabled>
    <id type="integer">1</id>
    <identifier>onapp-ojgg2jk75zfzmw</identifier>
    <ip>109.123.105.163</ip>
    <label>ds6</label>
    <local_hypervisor_id nil="true"/>
    <updated_at type="datetime">2012-05-25T10:51:21+00:00</updated_at>
    <zombie_disks_size type="integer">0</zombie_disks_size>
    <usage type="integer">188</usage>
  </data_store>
  ...,
</data_stores>

Where:

created_at — the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

data_store_size — the size of your data store shown in GB

id — the data store ID

label — the data store label

local_hypervisor_id — the ID of the compute resource using this Data Store

updated_at — the date when the Data Store was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format

data_store_group_id — the ID of a data store zone to which a particular data store is attached

zombie_disk_size — the size of zombie disks attached to this data store in GB.

enabled — true if a data store is enabled and you can attach disks to it, otherwise false.

### 25.20 Add Data Store Join to Compute Resource

To add a data store to the compute resource, use the following request:

POST /settings/hypervisors/:hypervisor_id/data_store_joins.xml

POST /settings/hypervisors/:hypervisor_id/data_store_joins.json

**XML Request Example**

```
curl -i -X POST
http://onapp.test/settings/hypervisors/14/data_store_joins.xml -d
```

**JSON Request Example**
This request creates a data store join, attaching a data store (`data_store_id*`) to a specified compute resource.

You can add data stores to a compute resource only if the zones to which the two entities belong are of the same type. For more information refer to [Zone Types](#).

### 25.21 Remove Data Store Join from Compute Resource

To remove a data store join from a compute resource, use the following request:

```
DELETE /settings/hypervisors/:hypervisor_id/data_store_joins/:id.xml
DELETE /settings/hypervisors/:hypervisor_id/data_store_joins/:id.json
```

**XML Request Example**

```
curl -i -X DELETE
http://onapp.test/settings/hypervisors/13/data_store_joins/365.xml
-u admin:password -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X DELETE
http://onapp.test/settings/hypervisors/13/data_store_joins/365.json
-u admin:password -H 'Accept: application/json' -H 'Content-type: application/json'
```

Returns HTTP 204 response on successful deletion, or HTTP 404 when a compute resource with the ID specified is not found, or the URL requested is incorrect.

### 25.22 Get List of Compute Resource Network Joins

To see the network joins of the compute resource, use the following request:

```
GET/settings/hypervisors/:hypervisor_id/network_joins.xml
GET/settings/hypervisors/:hypervisor_id/network_joins.json
```

**XML Request Example**

```
curl -i -X GET
-H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/settings/hypervisors/668/network_joins.xml
```

**JSON Request Example**

XML Output Example

```
<network_joins type="array">
<network_join>
  <id type="integer">668</id>
  <network_id type="integer">78</network_id>
  <interface>eth2</interface>
  <created_at type="dateTime">2017-03-05T12:28:33+02:00</created_at>
  <updated_at type="dateTime">2017-03-05T12:28:33+02:00</updated_at>
  <target_join_id type="integer">1</target_join_id>
  <target_join_type>Hypervisor</target_join_type>
</network_join>
</network_joins>
```

Where:

- **id** - the network join ID
- **network_id** - the ID of the assigned network
- **interface** - label of the network interface used to create a network join
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss] format
- **updated_at** - the date in the [YYYY][MM][DD][hh][mm][ss] format
- **target_join_id** - the ID of the join target; in this case it is the compute resource ID
- **target_join_type** - type of join target; in this case it is compute resource

25.23 Add Network Join to Compute Resource

To create a network join (assign the network to the compute resource), use the following request:

```
POST /settings/hypervisors/:hypervisor_id/network_joins.xml
POST /settings/hypervisors/:hypervisor_id/network_joins.json
```

You can add networks to a compute resource only if the zones to which the two entities belong are of the same type. For more information refer to Zone Types.

XML Request Example

```
curl -i -X POST
http://onapp.test/settings/hypervisors/35/network_joins.xml -d
'Content-type: application/xml'
```

JSON Request Example
curl -i -X POST
http://onapp.test/settings/hypervisors/35/network_joins.json -d
'{"network_join":{"network_id":"4", "interface":"interface_test"}}' -u
user:userpass -H 'Accept: application/json' -H 'Content-type:
application/json'

Where:

network_id* - ID of the network you wish to attach
interface* - the name of the appropriate network interface

25.24 Remove Network Join from Compute Resource

To remove a network join from a compute resource, use the following request:

DELETE /settings/hypervisors/:hypervisor_id/network_joins/:id.xml
DELETE /settings/hypervisors/:hypervisor_id/network_joins/:id.json

XML Request Example

curl -i -X DELETE
http://onapp.test/settings/hypervisors/15/network_joins/53.xml -u
user:userpass -H 'Accept: application/xml' -H 'Content-type:
application/xml'

JSON Request Example

curl -i -X DELETE
user:userpass -H 'Accept: application/json' -H 'Content-type:
application/json'

Returns HTTP 204 response on successful deletion or HTTP 404 when a resolver with the ID specified is not found, or the URL requested is incorrect.

25.25 Enable/Disable Open vSwitch

To enable the Open vSwitch, use the following request:

PUT http://onapp.test/settings/compute_resources/:compute
resource_id.xml
PUT http://onapp.test/settings/compute_resources/:compute
resource_id.json

XML Request Example

curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-
type: application/xml' -d '<compute_resource><ovs>1</ovs></compute
resource>' --url http://onapp.test/settings/compute_resources/15.xml

JSON Request Example
To disable the Open vSwitch, set the ovs parameter to 0. Returns HTTP 204 response on successful deletion, or HTTP 404 on failure.

25.26 Power Cycle CloudBoot Compute Resource

To power cycle a CloudBoot compute resource, use the following request:

POST /settings/hypervisors/:id/power_cycle.xml
POST /settings/hypervisors/:id/power_cycle.json

XML Request Example

```
curl -i -X POST http://onapp.test/settings/hypervisors/35/power_cycle.xml
-u user:userpass -H 'Accept: application/xml' -H 'Content-type:application/xml'
```

JSON Request Example

```
curl -i -X POST http://onapp.test/settings/hypervisors/35/power_cycle.json
-u user:userpass -H 'Accept: application/json' -H 'Content-type:application/json'
```

25.27 Get CPU Quota for Compute Resource

To view CPU Quota for compute resource, use the following request:

GET /settings/hypervisors/:hv_id/cpu_quota.xml
GET /settings/hypervisors/:hv_id/cpu_quota.json

XML Request Example

```
curl -i -X GET http://onapp.test/settings/hypervisors/35/cpu_quota.xml
-u admin:password -H 'Accept: application/xml' -H 'Content-type:application/xml'
```

JSON Request Example

```
curl -i -X GET http://onapp.test/settings/hypervisors/35/cpu_quota.json
-u admin:password -H 'Accept: application/json' -H 'Content-type:application/json'
```

XML Output Example
<cpu_quota>
   <enabled type="boolean">false</enabled>
   <value type="integer">0</value>
</cpu_quota>

Where:

*enabled* - true, if CPU Quota is enabled, otherwise, *false*

*value* - the value of CPU Quota. The maximum value is 99%.

### 25.28 Edit CPU Quota for Compute Resource

To edit CPU Quota for compute resource, use the following request:

**PUT** /settings/hypervisors/:hv_id/cpu_quota.xml

**PUT** /settings/hypervisors/:hv_id/cpu_quota.json

**XML Request Example**

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u admin:password -d '<cpu_quota><enabled>true</enabled><value>19</value></cpu_quota>' --url http://onapp.test/settings/hypervisors/14/cpu_quota.xml

**JSON Request Example**


**XML Output Example**

```xml
<cpu_quota>
   <enabled type="boolean">true</enabled>
   <value type="integer">19</value>
</cpu_quota>
```

Where:

*enabled* - true, if CPU Quota is enabled, otherwise, *false*

*value* - set the value of CPU Quota. The maximum value is 99%.

### 25.29 Enable Kernel Crash Dumping

To enable kernel crash dumping, use the following request:

**PUT** /settings/hypervisors/:hypervisor_id/crash_debug.xml

**PUT** /settings/hypervisors/:hypervisor_id/crash_debug.json

**XML Request Example**

```xml
<cpu_quota>
   <enabled type="boolean">true</enabled>
   <value type="integer">19</value>
</cpu_quota>
```
curl "http://onapp.test/settings/hypervisors/3/crash_debug.xml" -d
  '<hypervisor><crash_debug>true</crash_debug></hypervisor>' -X PUT \  
  -u user:pass \  
  -H "Accept: application/xml" \  
  -H "Content-Type: application/xml"

25.30 Enable Maintenance Mode for Xen/KVM Compute Resource

Maintenance mode is available for both Static and CloudBoot compute resources.

To enable maintenance mode for Xen/KVM compute resource, use the following request:

PUT http://onapp.test/settings/hypervisors/:hypervisor_id/maintenance_mode/enable.xml
PUT http://onapp.test/settings/hypervisors/:hypervisor_id/maintenance_mode/enable.json

XML Request Example

curl -X PUT
http://onapp.test/settings/hypervisors/14/maintenance_mode/enable.xml -d
  '<force>1</force>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

Where:

- **crash_debug** - true if the crash dumping option is enabled, otherwise, false

After you have enabled kernel crash dumping, reboot your compute resources at a convenient time to apply the changes.
curl -X PUT
http://onapp.test/settings/hypervisors/14/maintenance_mode/enable.json -d
'{"force":"1"}' -u user:userpass -H 'Accept: application/json' -H
'Content-type: application/json'

*force* - set 1 to enable maintenance node forcefully
Returns HTTP 204 response on a success, or HTTP 422 on failure.

### 25.31 Disable Maintenance Mode for Xen/KVM Compute Resource

- Maintenance mode is available for both Static and CloudBoot compute resources.
- Disabling maintenance mode initiates automatic compute resource reboot.

To disable maintenance mode for Xen/KVM compute resource, use the following request:

PUT http://onapp.test/settings/hypervisors/:hypervisor_id/maintenance_mode/disable.xml
PUT http://onapp.test/settings/hypervisors/:hypervisor_id/maintenance_mode/disable.json

**XML Request Example**

```bash
curl -X PUT
http://onapp.test/settings/hypervisors/13/maintenance_mode/disable.xml -u
user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -X PUT
http://onapp.test/settings/hypervisors/13/maintenance_mode/disable.json -u
user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Returns HTTP 204 response on a success, or HTTP 422 on failure.

### 25.32 Add Backup Server to Compute Resource

To create a backup server join (assign the backup server to the compute resource), use the following request:

POST /settings/hypervisors/:hypervisor_id/backup_server_joins.xml
POST /settings/hypervisors/:hypervisor_id/backup_server_joins.json
You can add a backup server to a compute resource only if the zones to which the two entities belong are of the same type. For more information refer to [Zone Types](#).

**XML Request Example**

```
```

**JSON request Example**

```
```

Where:

*backup_server_id* - ID of the backup server you wish to attach

You can check ID of the required backup server with GET method:

- GET /settings/backup_servers.xml
- GET /settings/backup_servers.json

### 25.33 Remove Backup Server from Compute Resource

To remove a backup server from the compute resource, use the following request:

DELETE /settings/hypervisors/:hypervisor_id/backup_server_joins/:id.xml

DELETE /settings/hypervisors/:hypervisor_id/backup_server_joins/:id.json

**XML Request Example**

```
curl -i -X DELETE http://onapp.test/settings/hypervisors/14/backup_server_joins/64.xml -u admin:password -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X DELETE http://onapp.test/settings/hypervisors/14/backup_server_joins/64.json -u admin:password -H 'Accept: application/json' -H 'Content-type: application/json'
```
You can check backup server join ID with GET method:
GET /settings/hypervisors/:hypervisor_id/backup_server_joins.xml
GET /settings/hypervisors/:hypervisor_id/backup_server_joins.json

25.34 Enable/Disable Compute Zone Custom Config

The compute zone custom config is applicable to CloudBoot, XEN, KVM, Baremetal, and Smart compute zones. If enabled, the compute zone custom config is applied before the compute resource custom config.

To enable or disable applying the compute zone custom config for a compute resource, use the following requests:

XML Request Example


JSON Request Example


Where:

apply_hypervisor_group_custom_config - set this parameter to 1 to enable applying the compute zone custom config, otherwise, set to 0

25.35 Power On Virtual Servers on Xen/KVM Compute Resource

To power on all virtual servers that run on the compute resource, use the following request:

POST /hypervisors/:hypervisor_id/virtual_machines/startup.xml
POST /hypervisors/:hypervisor_id/virtual_machines/startup.json

XML Request Example
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
  '<virtual_machines><virtual_machine>virtual_server_identifier_one</virtual_machine><virtual_machine>virtual_server_identifier_two</virtual_machine><shutdown_type>hard</shutdown_type></virtual_machines>'
  --url
  http://onapp.test/hypervisors/13/virtual_machines/stop.xml

Where:

virtual_machines - the array of virtual_machine parameters that include identifiers of VSs to be powered on
virtual_machine - the identifier of the VS to be powered on

Returns the 200 OK response if the request was completed successfully.

- Only the powered-off virtual will be powered on, while the already powered-on VSs will be skipped.
- The bulk power-on action is available only to virtual servers that are run on Xen and KVM compute resources.

25.36 Power Off Virtual Servers on Xen/KVM Compute Resource

To power off all virtual servers that run on the compute resource, use the following request:

POST /hypervisors/:hypervisor_id/virtual_machines/stop.xml
POST /hypervisors/:hypervisor_id/virtual_machines/stop.json

XML Request Example

curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
  '<virtual_machines><virtual_machine>virtual_server_identifier_one</virtual_machine><virtual_machine>virtual_server_identifier_two</virtual_machine><shutdown_type>hard</shutdown_type></virtual_machines>'
  --url
  http://onapp.test/hypervisors/13/virtual_machines/stop.xml

JSON Request Example

curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d
  '{"virtual_machines": ["virtual_server_identifier_one","virtual_server_identifier_two"]}'
  --url
  http://onapp.test/hypervisors/13/virtual_machines/stop.json

Where:

virtual_machines - the array of virtual_machine parameters that include identifiers of VSs to be powered off

virtual_machine - the identifier of the VS to be powered off

shutdown_type - specify the shut-down type that can be the following:

- graceful - to run a graceful shutdown of VSs
- hard - to run a forceful shutdown of VSs

Returns the 200 OK response if the request was completed successfully.

- Only the powered-on virtual will be powered off, while the already powered-off VSs will be skipped.
- The bulk power-off action is available only to virtual servers that are run on Xen and KVM compute resources.

25.37 Enable Storage Related Services for CloudBoot Compute Resources

If the storage-related services are enabled in System Configuration for the entire cloud, they are enabled for CloudBoot compute resources by default. You might want to enable the storage-related services for a CloudBoot compute resource in case they were earlier disabled.

To enable the storage-related services for a CloudBoot compute resource, use the following request:

POST /settings/hypervisors/:hypervisor_id/integrated_storage/enable.xml

POST /settings/hypervisors/:hypervisor_id/integrated_storage/enable.json

XML Request Example


JSON Request Example
25.38 Disable Storage Related Services for CloudBoot Compute Resources

To disable the storage-related services for a CloudBoot compute resource, use the following request:

```
POST /settings/hypervisors/:hypervisor_id/integrated_storage/disable.xml
POST /settings/hypervisors/:hypervisor_id/integrated_storage/disable.json
```

**XML Request Example**

```
curl -i -X POST
```

**JSON Request Example**

```
curl -i -X POST
```

Where:

- `force` - an optional parameter that you can set to `true` to disable the storage-related services if the health check has failed, otherwise, set to `false`

25.39 Edit Static Compute Resource Devices

To edit static compute resource devices, use the following request:

```
PUT /settings/hypervisors/:hypervisor_id/devices
```

**XML Request Example**

```
curl -i -X POST
```

**JSON Request Example**

```
curl -i -X POST
```
OnApp Cloud 6.3 API Guide

JSON Request Example

```bash
curl -i -X PUT http://onapp.test/settings/hypervisors/11/devices -d'
  "hypervisor_devices": {
    "586": {"format": "false", "status": "0"},
    "587": {"format": "false", "status": "0"},
    "588": {"format": "false", "status": "0"},
    "599": {"format": "false", "status": "2"},
    "593": {"status": "1"}
  },
  "hypervisor": {
    "cache_mirrors": "1",
    "cache_stripes": "1"
  },
  "hypervisor_id": "11"
' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

**hypervisor_devices** - an array of compute resource device details:
- **name** - the name of the disk
- **format** - set **true** to enable disk formatting, otherwise, **false**
- **status** - status of the disk:
  - 0 - unassigned
  - 1 - assigned to storage
  - 2 - assigned to cache

**hypervisor** - an array of compute resource details:
- **cache_mirrors** - the number of mirrors
- **cache_stripes** - the number of stripes

**hypervisor_id** - the ID of the compute resource

25.40 Get Details of Integrated Storage Settings

To view the details of integrated storage settings for a particular compute resource, use the following request:

GET /settings/hypervisors/:hypervisor_id/integrated_storage_settings.xml
GET /settings/hypervisors/:hypervisor_id/integrated_storage_settings.json

XML Request Example

```bash
curl -i -X GET -u user:userpass
http://onapp.test/settings/hypervisors/6/integrated_storage_settings.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example
curl -X GET -u user:userpass
http://onapp.test/settings/hypervisors/6/integrated_storage_settings.json
-H 'Accept: application/json' -H 'Content-type: application/json'

XML Output Example

```xml
<integrated_storage_settings>
  <bonding_mode>802.3ad</bonding_mode>
  <cache_mirrorsтип="integer">1</cache_mirrors>
  <cache_stripesтип="integer">1</cache_stripes>
  <controller_db_size тип="integer">128</controller_db_size>
  <controller_memory_size тип="integer">1024</controller_memory_size>
  <disks_per_controller тип="integer">4</disks_per_controller>
  <mtu тип="integer">1500</mtu>
  <vlan nil="integer">123</vlan>
</integrated_storage_settings>
```

Where:
- `bonding_mode` - the type of bonding mode
- `cache_mirrors` - the number of cache mirrors for the compute resource
- `cache_stripes` - the number of cache stripes for the compute resource
- `controller_db_size` - the controller DB size value
- `controller_memory_size` - the controller RAM value
- `disks_per_controller` - the number of disks per controller virtual server. By default, the controller virtual server is created per 4 disk drives.
- `mtu` - the maximum transportation unit size
- `vlan` - the ID of a VLAN number

25.41 Edit Integrated Storage Settings on Compute Resource

To edit integrated storage settings, use the following request:

PUT
/settings/hypervisors/:hypervisor_id/integrated_storage_settings.xml

PUT
/settings/hypervisors/:hypervisor_id/integrated_storage_settings.json

XML Request Example

```bash
curl -X PUT -u user:userpass http://onapp.test/settings/hypervisors/6/integrated_storage_settings.xml -d
  'answer' <integrated_storage_settings><bonding_mode>802.3ad</bonding_mode><cache_mirrors>1</cache_mirrors><cache_stripes>1</cache_stripes><controller_db_size>128</controller_db_size><controller_memory_size>1024</controller_memory_size><disks_per_controller>4</disks_per_controller><mtu>1500</mtu><vlan>123</vlan></integrated_storage_settings>''
```

JSON Request Example

Where:

- **bonding_mode** - the type of bonding mode
- **cache_mirrors** - the number of cache mirrors for the compute resource
- **cache_stripes** - the number of cache stripes for the compute resource
- **controller_db_size** - the controller DB size value
- **controller_memory_size** - the controller RAM value
- **disks_per_controller** - the number of disks per controller virtual server. By default, the controller virtual server is created per 4 disk drives.
- **mtu** - the maximum transportation unit size
- **vlan** - the ID of a VLAN number
26 Compute Zones

A compute zone consists of several compute resources sharing the same permissions and assigned to the same bucket. This class manages all the compute zones created in the cloud. Compute zones can have data stores and networks attached to them. The combination of compute resource, data store and network zones can be used to create private clouds and tiered services for customers. All API calls are available to this class.

26.1 Get List of Compute Zones

To get an array of compute zones set up within your cloud, use the following request:

GET /settings/hypervisor_zones.xml
GET /settings/hypervisor_zones.json

XML Request Example


JSON Request Example


XML Output Example
where:

created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

updated_at - the date when the compute zone was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

default_gateway - external gateway IP address used for the VMware utilization with the external firewall. All virtual machines within a compute zone will be rerouted to this gateway

vlan - address of a VLAN the default gateway is located on.

id - the compute zone ID

label* - title of a compute zone

location_group_id - ID of a location group the compute zone is assigned to

preconfigured_only - whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.

max_vms_start_at_once - the maximum number of virtual servers that can be started simultaneously within this compute zone

network_failure - true, if all compute resources in the compute zone failed

prefer_local_reads - 1 if the network throughput dependency for read heavy workloads, otherwise set 0. When the Use Local Read Path feature is enabled, reads go over the local software bridge to a local replica of the data rather than traverse a physical NIC + switch

recovery_type - specify the compute resource selection algorithm, which will be used on virtual server provisioning and recovery:

- roundrobin - set the roundrobin type to select the compute resource with maximum free RAM during the VS recovery
- `fillnext` - select the fillnext type to select the compute resource with minimum required free RAM. This option allows to fill compute resource as tightly as possible before starting to use next appliance in the zone.

`release_resource_type` - specify the release resource type. Release resource option allows to free up compute resource resources by over-committing RAM, CPU and CPU shares of virtual servers that are shut down.

- `memory_guarantee` - the actual free compute resource memory is calculated. All virtual servers residing on the compute resource will be able to start.

- `ballooning` - free compute resource memory is calculated with the ability to use memory over-committing. The ballooning option is only available for KVM compute resources. NOTE: Virtual server may be migrated to another compute resource if there is not enough memory for it to start up on the compute resource with the ballooning option enabled.

Do not use the ballooning option if there is at least one edge or storage server within the compute zone.

- `only_started_vms` - only the free memory of running virtual servers is calculated.

By default, the compute zone is created with the Memory Guarantee option enabled. In this case the release resources option is not used. Then, to enable resource over-committing you should choose either the Ballooning or Only Started VS option.

`failover_timeout` - time period for which the iterations will run during the failover.

`run-sysprep` - set 1 to enable Windows virtual server deployment without running sysprep.

`server_type` - specify the type of servers that will reside within this compute zone:

- `virtual` - choose the virtual type to create a Xen, KVM, VMware or CloudBoot zone
- `smart` - choose the smart server type to create a smart server zone
- `baremetal` - choose the baremetal server type to create a baremetal server zone

`storage_channel` - storage channel for the communication with the

`max_host_free_memory` - compute resource with maximum RAM value in this zone

`max_host_cpu` - compute resource with maximum RAM value in this zone

`cpu_units` - the number of CPU units applied to this zone

`cpu_flags_enabled` - true if CPU flags are enabled for the compute zone; otherwise false

`cpu_flags` - an array of CPU flags enabled for this compute zone

### Page History

v.4.2

- added the following parameter:
  - `preconfigured_only`
v.3.3

- added `cpu_units` parameter

v.3.1

- added the following parameters:
  - `server_type`
  - `release_resource_type`
  - `recovery_type`
  - `run_sysprep`
  - `failover_timeout`
  - `storage_channel`
  - `max_host_free_memory`
  - `max_host_cpu`

## 26.2 Get Compute Zone Details

To get the details for a particular compute zone, use the following request:

```plaintext
GET /settings/hypervisor_zones/:id.xml
GET /settings/hypervisor_zones/:id.json
```

### XML Request Example

```bash
```

### JSON Request Example

```bash
```

### XML Output Example
<hypervisor_group>
<closed type="boolean">false</closed>
<created_at type="datetime">2013-09-04T12:49:45+03:00</created_at>
<federation_enabled type="boolean">false</federation_enabled>
<federation_id nil="true"/>
<hypervisor_id nil="true"/>
<id type="integer">1</id>
<identifier nil="true"/>
<label>KVM C5 HV Zone</label>
<location_group_id type="integer">1</location_group_id>
<preconfigured_only type="boolean">true</preconfigured_only>
<server_type>virtual</server_type>
<traded type="boolean">false</traded>
<updated_at type="datetime">2015-06-11T17:56:12+03:00</updated_at>
<max_host_free_memory type="integer">1191</max_host_free_memory>
<max_host_cpu type="integer">4</max_host_cpu>
<prefer_local_reads type="boolean">false</prefer_local_reads>
<vlan nil="true"/>
<release_resource_type>ballooning</release_resource_type>
<network_failure type="boolean">false</network_failure>
<storage_channel type="integer">2</storage_channel>
<run_sysprep type="boolean">true</run_sysprep>
<default_gateway nil="true"/>
<recovery_type>roundrobin</recovery_type>
<failover_timeout type="integer">15</failover_timeout>
<cpu_units type="integer">1000</cpu_units>
<supplier_version nil="true"/>
<supplier_provider nil="true"/>
<cpu_flags_enabled type="boolean">true</cpu_flags_enabled>
<cpu_flags_type="array">...</cpu_flags>
</hypervisor_group>

Where:

created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at - the date when the compute zone was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
default_gateway - external gateway IP address used for the VMware utilization with the external firewall. All virtual machines within a compute zone will be rerouted to this gateway
closed - true, if federated zone is suspended by buyer, otherwise false
federation_id - the ID of a compute zone at the market
federation_enabled - true, if federation is enabled by seller, otherwise false
vlan - address of a VLAN the default gateway is located on.
hypervisor_id - the ID of a compute resource
id - the compute zone ID
label* - title of a compute zone
location_group_id - ID of a location group the compute zone is assigned to
preconfigured_only - whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.
traded - true, if the zone came from the Federation and was subscribed to by the user
max_vms_start_at_once - the maximum number of virtual servers that can be started simultaneously within this compute zone
network_failure - true, if all compute resources in the compute zone failed
`prefer_local_reads` - 1 if the network throughput dependency for read heavy workloads, otherwise set 0. When the Use Local Read Path feature is enabled, reads go over the local software bridge to a local replica of the data rather than traverse a physical NIC + switch.

`recovery_type` - specify the compute resource selection algorithm, which will be used on virtual server provisioning and recovery:

- `roundrobin` - set the roundrobin type to select the compute resource with maximum free RAM during the VS recovery.
- `fillnext` - select the fillnext type to select the compute resource with minimum required free RAM. This option allows to fill compute resource as tightly as possible before starting to use next appliance in the zone.

`release_resource_type` - specify the release resource type. Release resource option allows to free up compute resource resources by over-committing RAM, CPU and CPU shares of virtual servers that are shut down.

- `memory_guarantee` - the actual free compute resource memory is calculated. All virtual servers residing on the compute resource will be able to start.
- `ballooning` - free compute resource memory is calculated with the ability to use memory over-committing. The ballooning option is only available for KVM compute resources. NOTE: Virtual server may be migrated to another compute resource if there is not enough memory for it to start up on the compute resource with the ballooning option enabled.

Do not use the ballooning option if there is at least one edge or storage server within the compute zone.

- `only_started_vms` - only the free memory of running virtual servers is calculated.

By default, the compute zone is created with the Memory Guarantee option enabled. In this case the release resources option is not used. Then, to enable resource over-committing you should choose either the Ballooning or Only Started VS option.

`failover_timeout` - time period for which the iterations will run during the failover.

`run-sysprep` - set 1 to enable Windows virtual server deployment without running sysprep.

`server_type` - specify the type of servers that will reside within this compute zone:

- `virtual` - choose the virtual type to create a Xen, KVM, VMware or CloudBoot zone.
- `smart` - choose the smart server type to create a smart server zone.
- `baremetal` - choose the baremetal server type to create a baremetal server zone.

`storage_channel` - storage channel for the communication.

`max_host_free_memory` - compute resource with maximum RAM value in this zone.

`max_host_cpu` - compute resource with maximum RAM value in this zone.

`cpu_units` - the number of cpu units applied to a compute resource.

`cpu_flags` - an array of CPU flags enabled for this compute zone.
cpu_flags_enabled - true if CPU flags are enabled for the compute zone; otherwise false

Page History
v. 4.2
- added the following parameters:
  - preconfigured_only
  - cpu_flags
  - cpu_flags_enabled

v. 4.1
- added the following parameters:
  - closed
  - federation_id
  - federation_enabled
  - hypervisor_id
  - traded

v. 3.3
- added cpu_units parameter

v. 3.1
- added the following parameters:
  - server_type
  - release_resource_type
  - recovery_type
  - run_sysprep
  - failover_timeout
  - storage_channel
  - max_host_free_memory
  - max_host_cpu

26.3 Add Compute Zone
To add a new compute zone, use the following request:
POST /settings/hypervisor_zones.xml
POST /settings/hypervisor_zones.json

Note that the following requests will become deprecated in the future releases.
XML Request Example

```bash
curl -i -X POST http://onapp.test/settings/hypervisor_zones.xml -d "<hypervisor_group><label>test</label><server_type>virtual</server_type><location_group_id>38</location_group_id><preconfigured_only>true</preconfigured_only><release_resource_type>memory_guarantee</release_resource_type><max_vms_start_at_once>5</max_vms_start_at_once><recovery_type>roundrobin</recovery_type><failover_timeout>15</failover_timeout><run_sysprep>1</run_sysprep><default_gateway></default_gateway><vlan></vlan><cpu_units>1000</cpu_units><cpu_model_configuration>default</cpu_model_configuration><custom_config></custom_config></hypervisor_group>" -u user:userpass -H 'Accept: application/xml' -H 'Content-type:application/xml'
```

JSON Request Example

```bash
```

26.3.1 New API Requests

XML Request Example

```bash
curl -i -X POST http://onapp.test/settings/hypervisor_zones.xml -d "<hypervisor_group><label>test</label><server_type>virtual</server_type><location_group_id>38</location_group_id><preconfigured_only>true</preconfigured_only><release_resource_type>memory_guarantee</release_resource_type><max_vms_start_at_once>5</max_vms_start_at_once><recovery_type>roundrobin</recovery_type><failover_timeout>15</failover_timeout><run_sysprep>1</run_sysprep><default_gateway></default_gateway><vlan></vlan><cpu_units>1000</cpu_units><cpu_model_configuration>default</cpu_model_configuration><custom_config></custom_config></hypervisor_group>" -u user:userpass -H 'Accept: application/xml' -H 'Content-type:application/xml'
```

JSON Request Example

```bash
```

Where:

- **label** - title of a new compute zone
- **server_type** - specify the type of servers that will reside within this compute zone:
  - virtual - choose the virtual type to create a Xen, KVM, VMware or CloudBoot zone
  - smart - choose the smart server type to create a smart server zone
  - baremetal - choose the baremetal server type to create a baremetal server zone
  - vpc - choose the vpc server type to create a vCloud Director server zone
- **location_group** - specify the location group to which the compute zone will be assigned
- **preconfigured_only** - specify whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.
- **release_resource_type** - specify the release resource type. Release resource option allows to free up compute resources by over-committing RAM, CPU and CPU shares of virtual servers that are shut down. By default, the compute zone is created with the Memory Guarantee option enabled. In this case the release resources option is not used. Then, to enable resource over-committing you should choose either the Ballooning or Only Started VS option.
  - memory_guarantee - the actual free compute resource memory is calculated. All virtual servers residing on the compute resource will be able to start.
  - ballooning - free compute resource memory is calculated with the ability to use memory over-committing. The ballooning option is only available for KVM compute resources. NOTE: Virtual server may be migrated to another compute resource if there is not enough memory for it to start up on the compute resource with the ballooning option enabled.

**Do not use the ballooning option if there is at least one edge or storage server within the compute zone.**

- **only_started_vms** - only the memory of running virtual servers is calculated.
- **max_vms_start_at_once** - the maximum number of virtual servers that can be started simultaneously within this compute zone
- **recovery_type** - specify the compute resource selection algorithm, which will be used on virtual server provisioning and recovery, per compute zone:
  - roundrobin - set the roundrobin type to select the compute resource with maximum free RAM during the VS recovery
Note: this option behaves in different ways, depending on the event:

- On provisioning, the round-robin algorithm will be used on compute resource selection.
- On recovery, the compute resource with maximum free RAM will be selected.

- `fillnext` - select the fillnext type to select the compute resource with minimum required free RAM. This option allows to fill compute resource as tightly as possible before starting to use next appliance in the zone.

`failover_timeout` - time period for which the iterations will run during the failover if the compute resource does not respond.

`prefer_local_reads` - set 1 to minimise the network throughput dependency for read heavy workloads. When this option is enabled, reads go over the local software bridge to a local replica of the data rather than traverse a physical NIC + switch.

`run_sysprep` - set 1 to enable Windows virtual server deployment without running sysprep.

**NOTE:** It is not possible to set VS password when creating a Windows-based VMware virtual server without running a sysprep.

cpu_units - set the number of cpu units for applied to each compute resource in this compute zone.

cpu_model_configuration - select the CPU model for you compute zone (default, cpu_flags_enabled for CPU extended configuration, or host_passthrough_enabled for passthrough host CPU model)

custom_config - specify any custom commands you want to run when a compute zone is booted

**VMware parameters:**

default_gateway - external gateway IP address. All virtual servers within the compute zone will be rerouted to this gateway.

`vlan` - address of a VLAN, on which the default gateway is located, with prefix length specified. For example: 10.0.0.1/24.

**Page History**

v.6.3 Edge 1
- replaced the `cpu_flags_enabled` parameter with the `cpu_model_configuration` parameter

v.5.6
- added the `custom_config` parameter

v.4.2
- added the following parameters:
  - `preconfigured_only`
  - `cpu_flags_enabled`

v.3.3
- added **cpu_units** parameter

v.3.1

- added the following parameters:
  - `server_type`
  - `release_resource_type`
  - `recovery_type`
  - `run_sysprep`
  - `failover_timeout`

### 26.4 Edit Compute Zone

To edit an existing compute zone, use the following request:

```
PUT /settings/hypervisor_zones/:id.xml
PUT /settings/hypervisor_zones/:id.json
```

Note that the following requests will become deprecated in the future releases.

#### XML Request Example

```
curl -i -X PUT http://onapp.test/settings/hypervisor_zones/13.xml -d '...
```

#### JSON Request Example

```
curl -i -X PUT http://onapp.test/settings/hypervisor_zones/13.json -d '...
```
26.4.1 New API Requests

XML Request Example

curl -i -X PUT http://onapp.test/settings/hypervisor_zones/13.xml -d "<hypervisor_group><label>appliance</label><location_group_id>38</location_group_id><preconfigured_only>true</preconfigured_only><release_resource_type>memory_guarantee</release_resource_type><max_vms_start_at_once>10</max_vms_start_at_once><recovery_type>roundrobin</recovery_type><failover_timeout>20</failover_timeout><run_sysprep>1</run_sysprep><cpu_units>1000</cpu_units><update_cpu_units>1</update_cpu_units><cpu_guarantee>1</cpu_guarantee><cpu_model_configuration>default</cpu_model_configuration><custom_config></custom_config></hypervisor_group>" -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X PUT http://onapp.test/settings/hypervisor_zones/13.json -d '{"hypervisor_group":{"label":"hypervisor","location_group_id":"1","preconfigured_only":"true","release_resource_type":"memory_guarantee","max_vms_start_at_once":"55","recovery_type":"roundrobin","failover_timeout":"155","run_sysprep":"1","cpu_units":"1000","update_cpu_units":"1","cpu_guarantee":"1","cpu_model_configuration":"default","custom_config":""}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

Where:

- **label** - title of a new compute zone
- **server_type** - specify the type of servers that will reside within this compute zone:
  - **virtual** - choose the virtual type to create a Xen, KVM, VMware or CloudBoot zone
  - **smart** - choose the smart server type to create a smart server zone
  - **baremetal** - choose the baremetal server type to create a baremetal server zone
- **location_group** - edit the location group to which the compute zone is assigned. You can change the already set location if there are no virtual servers built on compute resources of this zone.
- **preconfigured_only** - specify whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs) built by setting resources manually creation.
- **release_resource_type** - specify the release resource type. Release resource option allows to free up compute resource resources by over-committing RAM, CPU and CPU shares of virtual servers that are shut down. By default, the compute zone is created with the Memory Guarantee option enabled. In this case the release resources option is not used. Then, to enable resource over-committing you should choose either the Ballooning or Only Started VS option.
  - **memory_guarantee** - the actual free compute resource memory is calculated. All virtual servers residing on the compute resource will be able to start.
  - **ballooning** - free compute resource memory is calculated with the ability to use memory over-committing. The ballooning option is only available for KVM compute resources.

NOTE: Virtual server may be migrated to another compute resource if there is not enough memory for it to start up on the compute resource with the ballooning option enabled.
Do not use the ballooning option if there is at least one edge or storage server within the compute zone.

**only_started_vms** - only the free memory of running virtual servers is calculated.

**max_vms_start_at_once** - the maximum number of virtual servers that can be started simultaneously within this compute zone.

**recovery_type** - specify the compute resource selection algorithm, which will be used on virtual server provisioning and recovery:

- **roundrobin** - set the *roundrobin* type to select the compute resource with maximum free RAM during the VS recovery. This option behaves in different ways, depending on the event:
  - On provisioning, the round-robin algorithm will be used on compute resource selection.
  - On recovery, the compute resource with maximum free RAM will be selected.

- **fillnext** - select the fillnext type to select the compute resource with minimum required free RAM. This option allows to fill compute resource as tightly as possible before starting to use next appliance in the zone.

**failover_timeout** - time period for which the iterations will run during the failover if the compute resource does not respond.

**prefer_local_reads** - set 1 to minimise the network throughput dependency for read heavy workloads. When this option is enabled, reads go over the local software bridge to a local replica of the data rather than traverse a physical NIC + switch.

**run-sysprep** - set 1 to enable Windows virtual server deployment without running sysprep.

**NOTE:** It is not possible to set VS password when creating a Windows-based VMware virtual server without running a sysprep.

**cpu_units** - set the amount of CPU units applied to the whole compute zone.

**update_cpu_units** - set 1 to update the amount of CPU units for each compute resource in this zone according to the cpu_units parameter.

**cpu_guarantee** - if 1, the system will make sure there is enough CPU in the compute zone to create a new VS.

**cpu_model_configuration** - select the CPU model for your compute zone ([default], cpu_flags_enabled for CPU extended configuration, or host_passthrough_enabled for passthrough host CPU model).

**custom_config** - specify any custom commands you want to run when a compute zone is booted.

**VMware parameters:**

**default_gateway** - external gateway IP address. All virtual servers within the compute zone will be rerouted to this gateway.

**vlan** - address of a VLAN, on which the default gateway is located, with prefix length specified. For example: 10.0.0.1/24.

**Page History**

v.6.3 Edge 1

- replaced the **cpu_flags_enabled** parameter with the **cpu_model_configuration** parameter.
v. 5.6
- added the custom_config parameter

v. 4.2
- added the following parameters:
  - preconfigured_only
  - cpu_flags_enabled

v. 3.3
- added the following parameters:
  - cpu_units
  - update_cpu_units
  - cpu_guarantee

v. 3.1
- added the following parameters:
  - server_type
  - release_resource_type
  - recovery_type
  - run_sysprep
  - failover_timeout

26.5 Delete Compute Zone

To delete a compute zone, use the following request:

```
DELETE /settings/hypervisor_zones/:id.xml
DELETE /settings/hypervisor_zones/:id.json
```

**XML Request Example**
```
```

**JSON Request Example**
```
```

You will get a 204 status response on success, and 404 if there is no such a compute zone with a requested ID or you entered incorrect URL.

26.6 Get List of Compute Resources Attached to Compute Zone

To get the list of compute resources attached to a compute zone, use the following request:
GET /settings/hypervisor_zones/:hypervisor_zone_id/hypervisors.xml
GET /settings/hypervisor_zones/:hypervisor_zone_id/hypervisors.json

XML Request Example

```
```

JSON Request Example

```
```

Returns the array of all compute resources attached to a particular compute zone.

### 26.7 Attach Compute Resource to Compute Zone

To attach compute resource to compute zone, use the following request:

XML Request Example

```
```

JSON Request Example

```
```

Where you have to specify ID of a compute zone and IDs of the compute resources you want to attach in the URL.

When you add a compute resource to a compute zone, it inherits the zone's type. It will be possible to move such a resource only to a compute zone of the same type. For more information refer to [Zone Types](#).

We strongly recommend that you avoid creating mixed compute zones:

- do not add CloudBoot and static boot compute resources to one compute zone
- do not add both XEN and KVM compute resources to one zone
The reason is that XEN VSs cannot migrate/failover to a KVM compute resource and KVM VSs cannot migrate/failover to a XEN compute resource.

26.8 Remove Compute Resource from Compute Zone

To remove compute resource from compute zone, use the following request:

POST /settings/hypervisor_zones/:hypervisor_zone_id/hypervisors/:hypervisor_id/detach.xml
POST /settings/hypervisor_zones/:hypervisor_zone_id/hypervisors/:hypervisor_id/detach.json

XML Request Example


JSON Request Example


Where you have to specify ID of a compute zone and IDs of the compute resources you want to remove in the URL.

26.9 Get List of Data Store Joins Attached to Compute Zone

To get the list of data store joins attached to a compute zone, use the following request:

GET /settings/hypervisor_zones/:hypervisor_zone_id/data_store_joins.xml
GET /settings/hypervisor_zones/:hypervisor_zone_id/data_store_joins.json

XML Request Example


JSON Request Example

XML Output Example

```xml
<data_store_joins type="array">
  <data_store_join>
    <created_at type="datetime">2011-01-17T13:16:31Z</created_at>
    <target_join_type>Hypervisor Group</target_join_type>
    <updated_at type="datetime">2011-01-17T13:16:31Z</updated_at>
    <data_store_id type="integer">2</data_store_id>
    <hypervisor_id type="integer">nil="true">nil</hypervisor_id>
    <id type="integer">7</id>
    <target_join_id type="integer">9</target_join_id>
  </data_store_join>
</data_store_joins>
```

Where:

- `created_at` - timestamp in DB when the record was created
- `updated_at` - timestamp in DB when the record was updated
- `target_join_type` - compute resource group for data store joins
- `data_store_id` - the ID of a data store attached to a compute zone
- `hypervisor_id` - the ID of an compute resource to which a data store is attached
- `id` - the data store join ID
- `target_join_id` - the ID of a compute zone for which a join is created

### 26.10 Add Data Store Join to Compute Zone

To add a data store join to a compute zone, use the following request:

**POST**

```
/setting
```

```xml
HTTP://onapp.test/settings/hypervisor_zones/35/data_store_joins.xml
```

**XML Request Example**

```bash
```

**JSON Request Example**

---

346
This request attaches a particular data store join (:data_store_id*) to a specific compute zone (:hypervisor_zone_id).
You can add data stores to a compute zone only if the data store belongs to a data store zone of the same type as the compute zone. For more information refer to Zone Types.

### 26.11 Remove Data Store Join from Compute Zone

To remove a data store join from a compute zone, use the following request:

```
curl -i -X DELETE
http://onapp.test/settings/hypervisor_zones/:hypervisor_zone_id/data_store_joins/:id.xml
```

**XML Request Example**

```
curl -i -X DELETE
http://onapp.test/settings/hypervisor_zones/24/data_store_joins/13.xml
```

**JSON Request Example**

```
curl -i -X DELETE
http://onapp.test/settings/hypervisor_zones/24/data_store_joins/13.json
```

Where:

- **id** – is an ID of data store join

Returns HTTP 204 response on successful deletion, or HTTP 404 when a data store join with the ID specified is not found, or the URL requested is incorrect.

### 26.12 Get List of Network Joins Attached to Compute Zone

To get the list of network joins attached to a compute zone, use the following request:

```
curl -i -X GET
http://onapp.test/settings/hypervisor_zones/:hypervisor_zone_id/network_joins.xml
```

**XML Request Example**

```
curl -i -X GET
http://onapp.test/settings/hypervisor_zones/24/network_joins.xml
```
An array of network joins is returned on successful request.

**XML Output Example**

```
<network_joins type="array">
  <network_join>
    <created_at type="datetime">2011-02-01T12:27:52Z</created_at>
    <network_id type="integer">1</network_id>
    <target_join_type>Hypervisor Group</target_join_type>
    <updated_at type="datetime">2011-02-01T12:27:52Z</updated_at>
    <hypervisor_id type="integer">6</hypervisor_id>
    <interface>eth2</interface>
    <target_join_id type="integer">1</target_join_id>
  </network_join>
</network_joins>
```

**Where:**

- created_at - the timestamp when the record was created
- network_id - the ID of a network attached to this zone
- target_join_type - compute resource group for a network join
- updated_at - the timestamp when the record was updated
- hypervisor_id - the ID of an compute resource to which this network is assigned
- id - the network join ID
- interface - the network join interface
- target_join_id - the ID of an compute zone to which this network join is attached

### 26.13 Add Network Join to Compute Zone

To add a network join to a compute zone, use the following request:

- POST /settings/hypervisor_zones/:hypervisor_zone_id/network_joins.xml
- POST /settings/hypervisor_zones/:hypervisor_zone_id/network_joins.json

You can add networks to a compute zone only if the network belongs to a network zone of the same type as the compute zone. For more information refer to [Zone Types](#).
### 26.14 Remove Network Join from Compute Zone

To remove a network join from a compute zone, use the following request:

```
DELETE /settings/hypervisor_zones/:hypervisor_zone_id/network_joins/:id.xml
DELETE /settings/hypervisor_zones/:hypervisor_zone_id/network_joins/:id.json
```

**XML Request Example**

```
curl -i -X DELETE
http://onapp.test/settings/hypervisor_zones/14/network_joins/35.xml
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X DELETE
http://onapp.test/settings/hypervisor_zones/14/network_joins/35.json
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Returns HTTP 204 response on successful deletion, or HTTP 404 when a network join with the ID specified is not found, or the URL requested is incorrect.

### 26.15 Update CPU Flags for Compute Zone

To update CPU flags for a compute zone, use the following request:

```
PUT /settings/hypervisor_zones/:hypervisor_zone_id/cpu_configuration.xml
```
PUT
/settings/hypervisor_zones/:hypervisor_zone_id/cpu_configuration.json

XML Request Example

```
curl -i -X PUT
http://onapp.test/settings/hypervisor_zones/14/cpu_configuration.xml -d
'"<hypervisor_group"><cpu_flags_enabled>true</cpu_flags_enabled>
<baseline_cpu_flags><baseline_cpu_flag>invtdsc</baseline_cpu_flag>
<baseline_cpu_flag>rdtscp</baseline_cpu_flag></baseline_cpu_flags>
<cpu_flags><cpu_flag>x2apic</cpu_flag><cpu_flag>ds</cpu_flag></cpu_flags></hypervisor_group>"
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X PUT
http://onapp.test/settings/hypervisor_zones/14/cpu_configuration.json -d
'"{"hypervisor_group":{"cpu_flags_enabled":"true","baseline_cpu_flags":{"invtsc","rdtscp"},"cpu_flags":{"x2apic","ds"}}}"
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- `cpu_flags_enabled` - true if CPU flags are enabled for the compute zone, otherwise, false
- `baseline_cpu_flags` - the array of additional CPU flags that are not enabled for the compute zone
- `cpu_flags` - the array of additional CPU flags that are enabled for the compute zone

Starting from the 5.7 version, it is not possible to edit the list of default CPU flags that are automatically attached to the compute zone when the Extended CPU Configuration option is enabled.

Page History
v. 5.7
- added the `baseline_cpu_flag` parameter

26.16 Get Extended CPU Configuration Details for Compute Zone

To get extended CPU configuration details for a compute zone, use the following request:

GET
/settings/hypervisor_zones/:hypervisor_zone_id/cpu_configuration.xml
GET
/settings/hypervisor_zones/:hypervisor_zone_id/cpu_configuration.json

XML Request Example
curl -i -u user:userpass -X GET
http://onapp.test/settings/hypervisor_zones/14/cpu_configuration.xml

**JSON Request Example**

curl -i -u user:userpass -X GET
http://onapp.test/settings/hypervisor_zones/14/cpu_configuration.json

**XML Output Example**

```xml
<cpu_configuration>
  <baseline_cpu_flags type="array">
    <baseline_cpu_flag>invtsc</baseline_cpu_flag>
  </baseline_cpu_flags>
  <cpu_flags type="array">
    <cpu_flag>invtsc</cpu_flag>
  </cpu_flags>
  <cpu_model>Nehalem</cpu_model>
  <default_cpu_flags type="array">
    <default_cpu_flag>Intel</default_cpu_flag>
  </default_cpu_flags>
</cpu_configuration>
```

Where:

- **baseline_cpu_flags** - the array of additional CPU flags that are not enabled for the compute zone
- **cpu_flags** - the array of additional CPU flags that are enabled for the compute zone
- **cpu_model** - the CPU model of the compute zone
- **default_cpu_flags** - the array of default CPU flags that are automatically enabled for the compute zone and cannot be edited

### 26.17 Add Backup Server to Compute Zone

To create a backup server join (assign the backup server to the compute zone), use the following request:

**POST**

```
/settings/hypervisor_zones/:hypervisor_zone_id/backup_server_joins.xml
```

**POST**

```
/settings/hypervisor_zones/:hypervisor_zone_id/backup_server_joins.json
```

You can add backup servers to a compute zone only if the the backup server belongs to a backup server zone of the same type as the compute zone. For more information refer to [Zone Types](#).

**XML Request Example**
26.18 Remove Backup Server from Compute Zone

To remove a backup server from the compute zone, use the following request:

**XML Request Example**
```bash
```

**JSON Request Example**
```bash
```

You can check backup server join ID with GET method:
OnApp Cloud 6.3 API Guide

GET /settings/hypervisors/:hypervisor_id/backup_server_joins.xml
GET /settings/hypervisors/:hypervisor_id/backup_server_joins.json

26.19 Edit Compute Zone Custom Config
To edit a custom config of a compute zone, use the following request:
XML Request Example
user:userpass -d
'<hypervisor_group><custom_config>custom_config</custom_config></hyperviso
r_group>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example
user:userpass -d '{"hypervisor_group":{"custom_config":"custom_config"}}'
-H 'Accept: application/json' -H 'Content-type: application/json'

Where:
custom_config - specify any custom commands that will be run when a compute zone is booted

26.20 Get List of Backup Resource Zones Attached to Compute
Zone
To get the list of backup resource zones attached to a compute zone, use the following
request:
GET
/settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones
.xml
GET
/settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones
.json
XML Request Example
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type:
application/xml' -u user:userpass --url
http://onapp.test/settings/hypervisor_zones/13/backups/resource_zones.xml

JSON Request Example

353


OnApp Cloud 6.3 API Guide

```bash
```

**XML Output Example**

```xml
<resource_zones type="array">
  <resource_zone>
    <created_at type="dateTime">2018-03-27T18:12:40+03:00</created_at>
    <id type="integer">1</id>
    <label>backup_resource_zone</label>
    <location_group_id>2</location_group_id>
    <updated_at type="dateTime">2018-03-28T14:58:33+03:00</updated_at>
  </resource_zone>
  ...
</resource_zones>
```

**Where:**

- `resource_zone` - the array of parameters for the backup resource zone
- `created_at` - the date when the backup resource zone was created in the [YYYY][MM][DD][hh][mm][ss] format
- `id` - the ID of the backup resource zone
- `label` - the label of the backup resource zone
- `location_group_id` - the ID of the location group to which the backup resource zone is added
- `updated_at` - the date when the backup resource zone was updated in the [YYYY][MM][DD][hh][mm][ss] format

**Page History**

v.5.9

- updated the API requests:
  - from GET
    /settings/compute_zones/:compute_zone_id/backups/resource_zones.xml to GET
    /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones.xml
  - from GET
    /settings/compute_zones/:compute_zone_id/backups/resource_zones.json to GET
    /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones.json

**26.21 Add Backup Resource Zone to Compute Zone**

To add a backup resource zone to a compute zone, use the following request:

**POST**

/settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:backup_zone_id.xml
POST
/settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:backup_zone_id.json

XML Request Example


JSON Request Example


Page History

v. 5.9
- updated the API requests:
  - from POST /settings/compute_zones/:compute_zone_id/backups/resource_zones/:backup_zone_id.xml to POST /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:backup_zone_id.xml
  - from POST /settings/compute_zones/:compute_zone_id/backups/resource_zones/:backup_zone_id.json to POST /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:backup_zone_id.json

26.22  Remove Backup Resource Zone from Compute Zone

To remove a backup resource zone from a compute zone, use the following request:

DELETE /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:resource_zone_id.xml

DELETE /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:resource_zone_id.json

XML Request Example


JSON Request Example

Page History
v. 5.9
- updated the API requests:
  - from DELETE /settings/compute_zones/:compute_zone_id/backups/resource_zones/:resource_zone_id.xml to DELETE /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:resource_zone_id.xml
  - from DELETE /settings/compute_zones/:compute_zone_id/backups/resource_zones/:resource_zone_id.json to DELETE /settings/hypervisor_zones/:hypervisor_group_id/backups/resource_zones/:resource_zone_id.json

26.23 Enable/Disable Failover

To enable failover, use the following request:

PATCH /settings/hypervisor_zones/:id/manage_failover.xml
PATCH /settings/hypervisor_zones/:id/manage_failover.json

XML Request Example

```
curl "http://onapp.test/settings/hypervisor_zones/7/manage_failover.xml" -d '<?hypervisor-group><failover-status type="boolean">false</failover-status></hypervisor-group>' -X PATCH \
-u user:userpass \
-H "Accept: application/xml" \
-H "Content-Type: application/xml"
```

JSON Request Example

```
curl "http://onapp.test/settings/hypervisor_zones/7/manage_failover.json" -d '{"hypervisor_group":{"failover_status":false}}' -X PATCH \
-u user:userpass \
-H "Accept: application/json" \
-H "Content-Type: application/json"
```

To disable failover, set the failover_status parameter to true.
27 Container Servers

Container servers in OnApp are based on a CoreOS template and deployed on compute resources. Container servers have their own root accounts, so that container server owners can fully control, configure and manage their servers.

We do not support container servers on CloudBoot compute resources running on CentOS 5.

27.1 Get List of All Container Servers

To get the list of all container servers, use the following request:

GET /container_servers.xml
GET /container_servers.json

XML Request Example

http://onapp.test/container_servers.xml

JSON Request Example

curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -u user:userpass -u user:userpass --url
http://onapp.test/container_servers.json

XML Output Example
<container_servers type="array">
  <container_server>
    <add_to_marketplace nil="true"/>
    <admin_note nil="true"/>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <autoscale_service nil="true"/>
    <booted type="boolean">true</booted>
    <built type="boolean">true</built>
    <cdboot type="boolean">false</cdboot>
    <cores_per_socket type="integer">0</cores_per_socket>
    <cpu_shares type="integer">100</cpu_shares>
    <cpu_sockets nil="true"/>
    <cpu_threads nil="true"/>
    <cpu_units type="integer">1000</cpu_units>
    <cpus type="integer">1</cpus>
    <created_at type="datetime">2016-09-30T11:45:00+03:00</created_at>
    <customer_network_id nil="true"/>
    <deleted_at nil="true"/>
    <draas_keys type="array"/>
    <draas_mode type="integer">0</draas_mode>
    <edge_server_type nil="true"/>
    <enable_autoscale nil="true"/>
    <firewall_notrack type="boolean">false</firewall_notrack>
    <hostname>sgsg</hostname>
    <hot_add_cpu nil="true"/>
    <hot_add_memory nil="true"/>
    <hypervisor_id type="integer"></hypervisor_id>
    <id type="integer">9255</id>
    <identifier>l4zz1458vgbeo3</identifier>
    <initial_root_password>n158758Aquhc2</initial_root_password>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <instance_package_id nil="true"/>
    <iso_id nil="true"/>
    <label>test</label>
    <local_remote_access_ip_address>111.177.7.33</local_remote_access_ip_address>
    <local_remote_access_port type="integer">5777</local_remote_access_port>
    <locked type="boolean">false</locked>
    <memory type="integer">512</memory>
    <min_disk_size type="integer">5</min_disk_size>
    <note nil="true"/>
    <operating_system>coreos</operating_system>
    <operating_system_distro>coreos</operating_system_distro>
    <preferred_hvs type="array"/>
    <recovery_mode nil="true"/>
    <remote_access_password>nmdf15487UpiO</remote_access_password>
    <service_password nil="true"/>
    <state>delivered</state>
    <storage_server_type nil="true"/>
    <strict_virtual_machine_id nil="true"/>
    <suspended type="boolean">false</suspended>
    <template_id type="integer">477</template_id>
    <template_label>Coreos current x64</template_label>
    <time_zone nil="true"/>
    <updated_at type="datetime">2016-10-21T14:13:17+03:00</updated_at>
    <user_id type="integer">758</user_id>
    <vapp_id nil="true"/>
    <vcenter_moref nil="true"/>
    <vcenter_url nil="true"/>
    <vmware_tools nil="true"/>
    <xen_id type="integer">992</xen_id>
    <ip_addresses type="array"/>
Where:

add_to_marketplace - empty for container servers; used for edge servers only

admin_note - an optional note of the administrator

allowed_hot_migrate - true if the template, on which the container server is based, supports hot migration; otherwise false

allowed_swap - true if swap disk is allowed (depends on the template the container server is based on); otherwise false

autoscale_service - currently, autoscaling is not available for container servers

booted - true if the container server is running, otherwise false

built - true if the container server is built, otherwise false

cores_per_socket - the amount of cores per socket

cpu_shares - CPU priority in percent's

cpu_sockets - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted

cpu_threads - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted

cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket

cpus - the number of allocated CPU cores
created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

customer_network_id - ID of a customer network

deleted_at - time when the container server was deleted

dev_ege_server_type - true if this is the edge server

dev_ege_server_type - true if autoscaling is allowed for this container server. Currently, autoscaling is not available for container servers.

dev_firewall_notrack - true if the NOTRACK rule is set in iptables

dev_hostname - the name of your host

dev_hot_add_cpu - true, if the CPU parameter can be changed without rebooting the container server, otherwise false

dev_hot_add_memory - true, if the memory parameter can be changed without rebooting the container server, otherwise false

dev_hypervisor_id - the ID of the compute resource used by this container server

dev_id - the container server ID

dev_identifier - the container server identifier

dev_initial_root_password - the container server root password

dev_initial_root_password_encrypted - true, if the root password is encrypted, otherwise false

dev_instance_package_id - ID of the instance package. Instance packages are not currently available for container servers.

dev_iso_id - the ID of the ISO the container server is based on

dev_label - the container server label

dev_local_remote_access_ip_address - IP address used for remote access

dev_local_remote_access_port - the port ID used for console access

dev_locked - true if the container server is locked; otherwise false

dev_memory - the RAM size allocated to this container server

dev_min_disk_size - the minimum disk size required to build a container server from a specified template

dev_note - an optional reminder for this container server made by a user account

dev_operating_system - operating system used by the container server

dev_operating_system_distro - the distribution of the OS from which this container server is built

dev_preferred_hvs - the array of preferable compute resources based on compute zone that meet some container server configuration settings

dev_recovery_mode - true if recovery mode allowed. Otherwise false

dev_remote_access_password - the password for the remote access

dev_service_password - service account password

dev_state - parameter reserved for future use

dev_storage_server_type - true if this is a storage server

dev_strict_virtual_machine_id - the ID of a container server that will never reside on the same compute resource with this container server

dev_suspended - true if container server is suspended, otherwise false

dev_template_id - the ID of the template the container server is based on

dev_template_label - the name of the template from which this container server is built
time_zone - the time zone set for the container server. This parameter is applicable only to Windows KVM and XEN servers.

updated_at - the date when the VS was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

user_id - the ID of a user assigned to this container server

vapp_id - this parameter is not applicable for container servers

vccenter_moref - this parameter is not applicable for container servers

vip - true if the container server has VIP status (gives migration priority)

vmware_tools - this parameter is not applicable for container servers

xen_id - the container server ID set by the virtualization engine

ip_addresses - an array of IP addresses assigned to this container server and their details:

  address - IP address

  broadcast - broadcast address

  created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

  customer_network_id - the ID of the customer VLAN the IP address belongs to

  disallowed_primary - true if not allowed to be used as primary, otherwise false

  gateway - gateway address

  hypervisor_id - the ID of a compute resource the IP address is associated with

  id - the ID of the IP address

  ip_address_pool_id - ID of the IP address pool the IP address is associated with

  network_address - the address of the network

  network_id - the ID of the network

  pxe - true, if this address can be used for cloudbooting a compute resource

  updated_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

  user_id - the ID of the user this IP address is assigned to

  free - true if free, otherwise false

  netmask - netmask for the IP address

monthly_bandwidth_used - container server monthly bandwidth in KB

total_disk_size - the total disk size in GB of all disks assigned to container server

support_incremental_backups - 1, if container server supports incremental backups, and 0 if it does not. Currently, backups are not available for container servers.

cpu_priority - this is a new parameter reserved for further use; currently will have the same value as cpu_shares

built_from_iso - true if the container server is built from ISO; otherwise false

acceleration - true if acceleration is enabled for the container server; otherwise false. Acceleration is not available for container servers.

acceleration_status - the status of acceleration: active or inactive. Acceleration is not available for container servers.

hypervisor_type - the type of the compute resource the container server is built on (for example: xen, kvm, vcloud, vmware)

price_per_hour - server's price per hour

price_per_hour_powered_off - price per hour when server is powered off
27.2 Get Container Server Details

To get the details of a particular container server, use the following request:

GET /container_servers/container_server_id.xml
GET /container_servers/container_server_id.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<container_server>
  <add_to_marketplace nil="true"/>
  <admin_note nil="true"/>
  <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
  <allowed_swap type="boolean">true</allowed_swap>
  <autoscale_service nil="true"/>
  <booted type="boolean">true</booted>
  <built type="boolean">true</built>
  <cboots type="boolean">false</cboots>
  <cores_per_socket type="integer">0</cores_per_socket>
  <cpu_shares type="integer">100</cpu_shares>
  <cpu_sockets nil="true"/>
  <cpu_threads nil="true"/>
  <cpu_units type="integer">1000</cpu_units>
  <cpus type="integer">1</cpus>
  <created_at type="datetime">2016-09-30T11:45:00+03:00</created_at>
  <customer_network_id nil="true"/>
  <deleted_at nil="true"/>
  <draas_keys type="array"/>
  <draas_mode type="integer">0</draas_mode>
  <enable_autoscale nil="true"/>
  <firewall_notrack type="boolean">false</firewall_notrack>
  <hostname>sgsg</hostname>
  <hot_add_cpu nil="true"/>
  <hot_add_memory nil="true"/>
  <hypervisor_id type="integer">1</hypervisor_id>
  <id type="integer">9234</id>
  <identifier>l4zz3h7458eo3</identifier>
  <initial_root_password>ny6325hc2</initial_root_password>
  <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
  <instance_package_id nil="true"/>
  <iso_id nil="true"/>
  <local_remote_access_ip_address>188.176.7.47</local_remote_access_ip_address>
  <local_remote_access_port type="integer">5777</local_remote_access_port>
  <locked type="boolean">false</locked>
  <memory type="integer">512</memory>
  <min_disk_size type="integer">5</min_disk_size>
  <note nil="true"/>
  <operating_system>coreos</operating_system>
  <operating_system_distro>coreos</operating_system_distro>
  <preferred_hvs type="array"/>
  <recovery_mode nil="true"/>
  <remote_access_password>nmdfve52145i0</remote_access_password>
  <service_password nil="true"/>
  <state>delivered</state>
  <strict_virtual_machine_id nil="true"/>
  <suspended type="boolean">false</suspended>
  <template_id type="integer">489</template_id>
  <time_zone nil="true"/>
  <updated_at type="datetime">2016-10-21T14:13:17+03:00</updated_at>
  <user_id type="integer">946</user_id>
  <vapp_id nil="true"/>
  <vcenter_moref nil="true"/>
  <vip nil="true"/>
  <vmware_tools nil="true"/>
  <xen_id type="integer">902</xen_id>
  <ip_addresses type="array"/>
</container_server>
<address>4.4.4.16</address>
<broadcast>4.4.4.266</broadcast>
<created_at type="datetime">2016-04-14T13:34:38+03:00</created_at>
<customer_network_id nil="true"/>
<disallowed_primary type="boolean">false</disallowed_primary>
<gateway>4.4.4.1</gateway>
<hypervisor_id nil="true"/>
<ip_address_pool_id nil="true"/>
<network_address>4.4.4.0</network_address>
<network_id type="integer">1</network_id>
<pxe type="boolean">false</pxe>
<updated_at type="datetime">2016-08-19T16:13:29+03:00</updated_at>
<brick_id nil="true"/>
<free type="boolean">false</free>
<netmask>255.255.255.0</netmask>
</ip_address>
</ip_addressess>
<monthly_bandwidth_used type="integer">77135</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>
</support_incremental_backups>
<cpu_priority type="integer">100</cpu_priority>
<built_from_iso type="boolean">false</built_from_iso>
<acceleration type="boolean">false</acceleration>
<acceleration_status>inactive</acceleration_status>
<cpu_units type="integer">6</cpu_units>
<price_per_hour type="float">500.0</price_per_hour>
<brick_id type="float">300.0</price_per_hour_powered_off>
</container_server>

Where:

add_to_marketplace - empty for container servers; used for edge servers only

admin_note - an optional note of the administrator

allowed_hot_migrate - true if the template, on which the container server is based, supports hot migration; otherwise false

allowed_swap - true if swap disk is allowed (depends on the template the container server is based on); otherwise false

autoscale_service - currently, autoscaling is not available for container servers

booted - true if the container server is running, otherwise false

built - true if the container server is built, otherwise false

cores_per_socket - the amount of cores per socket

cpu_shares - CPU priority in percent's

cpu_sockets - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted

cpu_threads - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted

cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket

cpus - the number of allocated CPU cores

created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

customer_network_id - ID of a customer network
deleted_at - time when the container server was deleted
edge_server_type - true if this is the edge server
enable_autoscale - true if autoscaling is allowed for this container server. Currently, autoscaling is not available for container servers.
firewall_notrack - true if the NOTRACK rule is set in iptables
hostname - the name of your host
hot_add_cpu - true, if the CPU parameter can be changed without rebooting the container server, otherwise false
hot_add_memory - true, if the memory parameter can be changed without rebooting the container server, otherwise false
hypervisor_id - the ID of the compute resource used by this container server
id - the container server ID
identifier - the container server identifier
initial_root_password - the container server root password
initial_root_password_encrypted - true, if the root password is encrypted, otherwise false
instance_package_id - ID of the instance package. Instance packages are not currently available for container servers.
iso_id - the ID of the ISO the container server is based on
label - the container server label
local_remote_access_ip_address - IP address used for remote access
local_remote_access_port - the port ID used for console access
locked - true if the container server is locked; otherwise false
memory - the RAM size allocated to this container server
min_disk_size - the minimum disk size required to build a container server from a specified template
note - an optional reminder for this container server made by a user account
operating_system - operating system used by the container server
operating_system_distro - the distribution of the OS from which this container server is built
preferred_hvs - the array of preferable compute resources based on compute zone that meet some container server configuration settings
recovery_mode - true if recovery mode allowed. Otherwise false
remote_access_password - the password for the remote access
service_password - service account password
state - parameter reserved for future use
storage_server_type - true if this is a storage server
strict_virtual_machine_id - the ID of a container server that will never reside on the same compute resource with this container server
suspended - true if container server is suspended, otherwise false
template_id - the ID of the template the container server is based on
template_label - the name of the template from which this container server is built
time_zone - the time zone set for the container server. This parameter is applicable only to Windows KVM and XEN servers.
updated_at - the date when the VS was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
user_id - the ID of a user assigned to this container server
vapp_id - this parameter is not applicable for container servers
vcenter_moref - this parameter is not applicable for container servers
vip - true if the container server has VIP status (gives migration priority)
vmware_tools - this parameter is not applicable for container servers
xen_id - the container server ID set by the virtualization engine

ip_addresses - an array of IP addresses assigned to this container server and their details:
  address - IP address
  broadcast - broadcast address
  created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
  customer_network_id - the ID of the customer VLAN the IP address belongs to
disallowed_primary - true if not allowed to be used as primary, otherwise false
gateway - gateway address
hypervisor_id - the ID of a compute resource the IP address is associated with
id - the ID of the IP address
ip_address_pool_id - ID of the IP address pool the IP address is associated with
network_address - the address of the network
network_id - the ID of the network
pxe - true, if this address can be used for cloudbooting a compute resource
updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
user_id - the ID of the user this IP address is assigned to
free - true if free, otherwise false
netmask - netmask for the IP address
monthly_bandwidth_used - container server monthly bandwidth in KB
total_disk_size - the total disk size in GB of all disks assigned to container server
support_incremental_backups - 1, if container server supports incremental backups, and 0 if it does not. Currently, backups are not available for container servers.
cpu_priority - this is a new parameter reserved for further use; currently will have the same value as cpu_shares
built_from_iso - true if the container server is built from ISO; otherwise false
acceleration - true if acceleration is enabled for the container server ; otherwise false. Acceleration is not available for container servers.
acceleration_status - the status of acceleration: active or inactive. Acceleration is not available for container servers.
hypervisor_type - the type of the compute resource the container server is built on (for example: xen, kvm, vcloud, vmware)
price_per_hour - server's price per hour
price_per_hour_powered_off - price per hour when server is powered off
27.3 Get Statuses for all Container Servers

To get statuses for all container servers, use the following request:

GET /container_servers/status.xml
GET /container_servers/status.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<container_servers type="array">
  <container_server>
    <id type="integer">1</id>
    <identifier>oku1sief887rqm</identifier>
    <hostname>v1.test</hostname>
    <template_id type="integer">1</template_id>
    <built type="boolean">true</built>
    <locked type="boolean">false</locked>
    <booted type="boolean">true</booted>
    <operating_system>linux</operating_system>
    <suspended type="boolean">false</suspended>
    <enable_autoscale type="boolean">true</enable_autoscale>
    <state>new</state>
  </container_server>
  ...
</container_servers>
```

Where:

- **id** - container server ID
- **identifier** — the container server identifier
- **hostname** — the name of your host
- **template_id** — the ID of the template the container server is based on
- **built** — true if the container server is built, otherwise false
- **locked** — true if the container server is locked; otherwise false
- **booted** — true if the container server is running, otherwise false
- **operating_system** — operating system used by the container server
- **suspended** — true if container server is suspended, otherwise false
- **enable_autoscale** — true if autoscaling is allowed for this container server
- **state** – container server state
27.4 Get Container Server Status

This parameter has been added in the 3.1 version.

To get status for a particular container server, use the following request:

GET /container_servers/:container_server_id/status.xml
GET /container_servers/:container_server_id/status.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```xml
<container_servers type="array">
  <container_server>
    <id type="integer">48</id>
    <identifier>b266b5h5et39iy</identifier>
    <hostname>qaaoxp</hostname>
    <template_id type="integer">111</template_id>
    <built type="boolean">true</built>
    <locked type="boolean">false</locked>
    <booted type="boolean">true</booted>
    <operating_system>windows</operating_system>
    <suspended type="boolean">false</suspended>
    <enable_autoscale type="boolean">false</enable_autoscale>
    <state>new</state>
  </container_server>
</container_servers>
```

Where:

- **id** - container server ID
- **identifier** — the container server identifier
- **hostname** — the name of your host
- **template_id** — the ID of the template the container server is based on
- **built** — true if the container server is built, otherwise false
- **locked** — true if the container server is locked; otherwise false
- **booted** — true if the container server is running, otherwise false
- **operating_system** — operating system used by the container server
suspended — true if container server is suspended, otherwise false
enable_autoscale — true if autoscaling is allowed for this container server
state — container server state

### 27.5 Get Container Server Cloud Config

To get a container server cloud config, use the following request:

GET /container_servers/:container_server_id/cloud_config.xml
GET /container_servers/:container_server_id/cloud_config.json

#### XML Request Example

```bash
```

#### JSON Request Example

```bash
```

#### XML Output Example
Where:

`cloud_config` - the cloud-config file, which enables you to customize different OS elements, such as network configuration, user accounts, etc. For more information refer to the Container Server Cloud Config document.

### 27.6 Add Container Server

To add a container server, use the following request:

```
POST /container_servers.xml
POST /container_servers.json
```

**XML Request Example**

```
<container_server>
  <cloud_config>
    write-files:
      - path: /etc/hosts
        permissions: '0644'
        content: |
          111.222.33.444 master1 coreos00
          555.666.77.888 master2 coreos01
    coreos:
      etcd2:
        name: master2
        initial-cluster:
          master1=http://111.222.33.444:2380,master2=http://555.666.77.888:2380
          initial-advertise-peer-urls: http://$public_ipv4:2380
          listen-client-urls: http://0.0.0.0:2379,http://0.0.0.0:4001
          fleet:
            public-ip: $public_ipv4
            metadata: "role=master"
            flannel:
              interface: $public_ipv4
              units:
                - name: etcd2.service
                  command: start
                - name: fleet.service
                  command: start
                - name: flanneld.service
                  command: start
  </cloud_config>
</container_server>
```
### JSON Request Example

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"container_server": 
  "hypervisor_group_id": "6", "hypervisor_id": "4", 
  "hostname": "my_container", 
  "domain": "localdomain", "primary_network_group_id": "11", 
  "required_virtual_machine_build": "1", "memory": "512", "label": "zaza TEST", 
  "cpus": "1", "swap_disk_min_iops": "100", "data_store_group_swap_id": "13", 
  "rate_limit": "0", "cpu_shares": "1", "enable_autoscale": "False", 
  "template_id": "742", "primary_disk_min_iops": "100", 
  "initial_root_password": "password", "selected_ip_address": "5.1.1.12", 
  "data_store_group_primary_id": "13", "primary_disk_size": "5", 
  "required_ip_address_assignment": "1", "swap_disk_size": "1"}'} --url http://onapp.test/container_servers.json
```

Where:

- **hypervisor_group_id**: the ID of the compute zone in which the container server will be created
- **hypervisor_id**: the ID of a compute resource where the container server will be built
- **hostname**: container server hostname
- **domain**: specify the domain for this VS. The default value is `localdomain`. This parameter is not applicable for Windows virtual servers.
- **primary_network_group_id**: the ID of the primary network group
- **required_virtual_machine_build**: set 1 to build container server automatically
- **memory**: amount of RAM assigned to the container server
- **label**: name of the container server
- **cpus**: number of CPUs assigned to the container server
- **swap_disk_min_iops**: minimum number of IO operations per second for swap disk (this is a SolidFire related parameter)
- **data_store_group_swap_id**: set the ID of the data store zone to which this swap disk is allocated
- **rate_limit**: set max port speed in Mbps or set 0 to get maximum port speed allowed by your bucket. If this parameter is omitted or sent without value, the default port speed will be configured for the container server. The default port speed depends on the maximum port speed set in your bucket and the **Max network interface port speed parameter** at Control.
Panel > Settings > Configuration. The system identifies which of the two values (in the bucket or in the configuration) is lower and sets it as the default port speed during container server creation.

cpu_shares - for KVM compute resource the CPU priority value is always 100. For XEN, set a custom value. The default value for XEN is 1.

enable_autoscale - true if autoscaling is allowed for this container server. Autoscaling is not currently available for container servers.

template_id - the ID of the template the container server is based on

primary_disk_min_iops - minimum number of I/O operations per second for primary disk (this is a SolidFire related parameter)

initial_root_password - the root password for a container server. Optional, if none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _ ], and the following special characters: -=!@#$'`\{}[]. You can use both lower- and uppercase letters.

selected_ip_address - an IP address to assign to this container server; if the parameter required_ip_address_assignment was set "1" but this parameter selected_ip_address is empty - the first available IP address will be assigned to container server automatically

data_store_group_primary_id - set the ID of the data store zone to which this primary disk is allocated

primary_disk_size - set the disk space for this container server

required_ip_address_assignment - set "1" if you want container server to be created with already assigned IP address, otherwise set "0"; IP address can be assigned after container server creation.

swap_disk_size - set swap space

Page History

v.5.4

- added the following parameters:
  - domain
  - selected_ip_address

- removed selected_ip_address_id parameter

27.7 Add Container Server Cloud Config

To add a container server cloud config, use the following request:

PATCH /container_servers/:container_server_id/cloud_config.xml
PATCH /container_servers/:container_server_id/cloud_config.json

XML Request Example


JSON Request Example
**27.7 Add Cloud Config**

To add the cloud-config file, use the following request:

```
curl -i -X PATCH -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"container_server": {"cloud_config": "user's cloud_config"}}' --url http://onapp.test/container_servers/13/cloud_config.json
```

Where:

*cloud_config* - add the cloud-config file, which enables you to customize different OS elements, such as network configuration, user accounts, etc.. For more information refer to the Container Server Cloud Config document.

**27.8 View Encrypted Container Server Password**

If the container server was created with password encryption enabled, you can view the password via API request (the request returns the decrypted password).

To view the encrypted password, use the following request:

GET /container_servers/:id/with_decrypted_password.xml
GET /container_servers/:id/with_decrypted_password.json

**XML Request Example**

```
curl -X GET -u user:userpass
http://onapp.test/container_servers/2/with_decrypted_password.xml?initial_root_password_encryption_key=encryptionkey
```

**JSON Request Example**

```
curl -X GET -u user:userpass
http://onapp.test/container_servers/2/with_decrypted_password.json?initial_root_password_encryption_key=encryptionkey
```

Where:

*id* – the container server's ID

**27.9 Build or Rebuild Container Server**

To build or rebuild a container server, use the following request:

POST /container_servers/:container_server_id/build.xml
POST /container_servers/:container_server_id/build.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where:

*template_id* - the ID of a template from which a container server should be built.

*required_startup* - set to 1 if you wish to start a container server after it is built. Otherwise set to 0.

*initial_root_password_encryption_key* - specify the password encryption passphrase for Windows templates you should specify the licensing type:

*licensing_type* - the type of a license: mak, kms or user own license

*licensing_key* - the key of a license, required if you have selected OWN licensing type, and not required for MAK and KMS licensing types

*licensing_server_id* - the ID of a template group where the KMS server details are indicated and to which the template belongs (either directly or through the child group)

### 27.10 Edit Container Server

To edit a container server, use the following request:

PUT /container_servers/:container_server_id.xml
PUT /container_servers/:container_server_id.json

**XML Request Example**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:pass -d '<container_server><hypervisor_group_id>6</hypervisor_group_id><hypervisor_id>4</hypervisor_id><primary_network_group_id>11</primary_network_group_id><required_virtual_machine_build>1</required_virtual_machine_build><memory>512</memory><label>TEST</label><cpus>1</cpus><swap_disk_min_iops>100</swap_disk_min_iops><data_store_group_swap_id>13</data_store_group_swap_id><rate_limit>0</rate_limit><cpu_shares>1</cpu_shares><enable_autoscale>False</enable_autoscale><template_id>742</template_id><primary_disk_min_iops>100</primary_disk_min_iops><initial_root_password_encryption_key>password</initial_root_password_encryption_key><selected_ip_address_id>None</selected_ip_address_id><data_store_group_primary_id>13</data_store_group_primary_id><primary_disk_size>5</primary_disk_size><required_ip_address_assignment>1</required_ip_address_assignment><swap_disk_size>1</swap_disk_size></container_server>' --url http://onapp.test/container_servers/14.xml
```

**JSON Request Example**

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:pass -d '{"container_server": {"primary_network_group_id": 11, "primary_disk_size": 5, "template_id": 742, "primary_disk_min_iops": 100, "memorial": 512, "label": "TEST", "primary_disk_size": 5, "required_ip_address_assignment": 1, "swap_disk_size": 1, "primary_disk_min_iops": 100, "initial_root_password": "password"}}' --url http://onapp.test/container_servers/14.json
```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"container_server": {"hypervisor_group_id":"6", "hypervisor_id":"4", "primary_network_group_id":"11", "required_virtual_machine_build":"1", "memory":"512", "label":"zaza TEST", "cpus":"1", "swap_disk_min_iops":"100", "data_store_group_swap_id":"13", "rate_limit":"0", "cpu_shares":"1", "enable_autoscale":"false", "template_id":"742", "primary_disk_min_iops":"100", "initial_root_password":"password", "selected_ip_address_id":"None", "data_store_group_primary_id":"13", "primary_disk_size":"5", "required_ip_address_assignment":"1", "swap_disk_size":"1"}}' --url http://onapp.test/container_servers/14.json

Where:

- **hypervisor_group_id**: the ID of the compute zone in which the container server will be created
- **hypervisor_id**: the ID of a compute resource where the container server will be built
- **primary_network_group_id**: the ID of the primary network group
- **required_virtual_machine_build**: set to 1 to build container server automatically
- **memory**: amount of RAM assigned to the container server
- **label**: name of the container server
- **cpus**: number of CPUs assigned to the container server
- **swap_disk_min_iops**: minimum number of IO operations per second for swap disk (this is a SolidFire related parameter)
- **data_store_group_swap_id**: set the ID of the data store zone to which this swap disk is allocated
- **rate_limit**: set max port speed in Mbps or set 0 to get maximum port speed allowed by your bucket. If this parameter is omitted or sent without value, the default port speed will be configured for the container server. The default port speed depends on the maximum port speed set in your bucket and the **Max network interface port speed parameter** at Control Panel > Settings > Configuration. The system identifies which of the two values (in the bucket or in the configuration) is lower and sets it as the default port speed during container server creation.
- **cpu_shares**: for KVM compute resource the CPU priority value is always 100. For XEN, set a custom value. The default value for XEN is 1.
- **enable_autoscale**: true if autoscaling is allowed for this container server. Autoscaling is not currently available for container servers.
- **template_id**: the ID of the template the container server is based on
- **primary_disk_min_iops**: minimum number of IO operations per second for primary disk (this is a SolidFire related parameter)
- **initial_root_password**: the root password for a container server. Optional, if none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _ ], and the following special characters: ~ ! @ # $ * _ + = ` \ { [ ] : ; ' , . ? / . You can use both lower- and uppercase letters.
- **selected_ip_address_id**: specify ID of an IP address to assign to this container server; if the parameter required_ip_address_assignment was set to "1" but this parameter selected_ip_address_id is empty - the first available IP address will be assigned to container server automatically
- **data_store_group_primary_id**: set the ID of the data store zone to which this primary disk is allocated
- **primary_disk_size**: set the disk space for this container server
required_ip_address_assignment - set "1" if you want container server to be created with already assigned IP address, otherwise set "0"; IP address can be assigned after container server creation.

swap_disk_size - set swap space

### 27.11 Edit Container Server Cloud Config

To edit a container server cloud config, use the following request:

PUT /container_servers/:container_server_id/cloud_config.xml
PUT /container_servers/:container_server_id/cloud_config.json

**XML Request Example**

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '"<container_server><cloud_config><user' cloud_config></cloud_config></container_server>" --url http://onapp.test/container_servers/13/cloud_config.xml

**JSON Request Example**

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"container_server": {"cloud_config": "user's cloud_config"}}' --url http://onapp.test/container_servers/13/cloud_config.json

**Where:**

cloud_config - edit the cloud-config file, which enables you to customize different OS elements, such as network configuration, user accounts, etc.. For more information refer to the Container Server Cloud Config document.

### 27.12 Change Container Server Owner

To reassign a container server to another user, use the following request:

POST /container_servers/:container_server_id/change_owner.xml
POST /container_servers/:container_server_id/change_owner.json

**XML Request Example**


**JSON Request Example**

Where:

user_id* – input ID of a new container server owner

custom_recipes_action - select one of the following options for container server's recipes:

- none - recipes owner will not be changed
- move - recipes owner will be changed
- copy - recipes will be copied to new container servers owner

backups_action - select one of the following options for container server's backups:

- none - backup owner will not be changed
- move - backup owner will be changed

- Instead of container server ID (:container_server_id) you may use container server identifier (:container_server_identifier).
- If container server can not be reassigned to another user, you will get an error message: "New owner has reached his backup creation limit or doesn't have enough disk space."

27.13 Reset Container Server Root Password

To reset the container server root password, use the following request:

POST /container_servers/:container_server_id/reset_password.xml
POST /container_servers/:container_server_id/reset_password.json

XML Request Example


JSON Request Example

Where:
container_server_id * - id of the container server, for which you want to reset password.

initial_root_password - the new root password for a container server. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _ ]. You can use both lowercase and uppercase letters.

The following characters are not allowed for Windows-based virtual servers:
- percent sign [%]
- double quotation marks ['"]
- brackets [,<,>]
- vertical bar [ ]
- caret ['^']
- ampersand ['&']
- parentheses [(,)]

initial_root_password_encryption_key - specify the password encryption passphrase.

You can also reset a container server password using the OnApp 2.3.2 API request:

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:
container_server_id * - id of the container server, for which you want to reset password.

### 27.14 Migrate Container Server

To migrate a container server to another compute resource, use the following request:

POST /container_servers/:container_server_id/migrate.xml
POST /container_servers/:container_server_id/migrate.json

**XML Request Example**
OnApp Cloud 6.3 API Guide

27.15 Set VIP Status for Container Server

To set/remove VIP status for a container server, use the following request:

POST /container_servers/:id/set_vip.xml
POST /container_servers/:id/set_vip.json

XML Request Example

```
```

JSON Request Example

```
```

Where:

destination* - the ID of a target compute resource where you migrate a container server
cold_migrate_on_rollback - set to 1 if you wish to switch to a cold migration if hot migration fails, otherwise set 0.

27.16 Delete Container Server

To delete a container server, use the following request:

DELETE /container_servers/:container_server_id.xml
DELETE /container_servers/:container_server_id.json

XML Request Example

```
```

JSON Request Example

```
```

Where:

vip - whether VIP status is enabled for the server or not. Set this parameter to 'true' to enable and to 'false' to disable the VIP status.
When you start up a container server, it might be implicitly cold migrated if the current compute resource does not have sufficient resources. For more information, refer to Virtual Server Provisioning.

To start up a container server, use the following request:

POST /container_servers/:container_server_id/startup.xml
POST /container_servers/:container_server_id/startup.json

**XML Request Example**

curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/32/startup.xml

**JSON Request Example**

curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/32/startup.json

You can also start up a container server in recovery mode. For this, run the following request:

**XML Request Example**

curl -i -X POST -u user:userpass -d '<mode>recovery</mode>' --url http://onapp.test/container_servers/32/startup.xml

**JSON Request Example**

curl -i -X POST -u user:userpass -d '{"mode":"recovery"}' --url http://onapp.test/container_servers/32/startup.json
27.18 Segregate Container Server

To segregate a container server (that is, instruct it never to reside on the same compute resource as another container server), use the following request:

PUT /container_servers/:container_server_id/segregation.xml
PUT /container_servers/:container_server_id/segregation.json

XML Request Example

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<container_server><strict_container_server_id>123</strict_container_server_id></container_server>'
  --url http://onapp.test/container_servers/23/segregation.xml
```

JSON Request Example

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"container_server":{"strict_container_server_id":"123"}}'
  --url http://onapp.test/container_servers/23/segregation.json
```

Where:

`strict_container_server_id`* - the ID of container server you wish to segregate from the given container server

27.19 Desegregate Container Server

To desegregate a container server (that is, cancel the instruction for it to never reside on the same compute resource as another container server), send an empty identifier using the following request:

DELETE /container_servers/:container_server_id/segregation.xml
DELETE /container_servers/:container_server_id/segregation.json

XML Request Example

```
curl -i -X DELETE -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<container_server><strict_container_server_id>123</strict_container_server_id></container_server>'
  --url http://onapp.test/container_servers/23/segregation.xml
```

JSON Request Example

```
curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"container_server":null}'
  --url http://onapp.test/container_servers/23/segregation.json
```

Where:

strict_container_server_id* - the ID of container server you wish to desegregate from the given container server

27.20 Reboot Container Server

To reboot a container server, use the following request:

POST /container_servers/:container_server_id/reboot.xml
POST /container_servers/:container_server_id/reboot.json

XML Request Example

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/34/reboot.xml
```

JSON Request Example

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/34/reboot.json
```

An HTTP 201 response is returned on a successful reboot. Unsuccessful reboot responses include HTTP 404 (resource not found – e.g. if the VS isn’t online) and HTTP 422 (request cannot be processed – for example, if parameters were incorrect).

27.21 Reboot Container Server in Recovery

To reboot a container server in recovery mode with a temporary login (“root”) and password (“recovery”), use the following request:

POST /container_servers/:container_server_id/reboot.xml
POST /container_servers/:container_server_id/reboot.json

XML Request Example

```bash
```

JSON Request Example

```bash
```
27.22 Boot Container Server from ISO

To boot container servers that are powered off from an ISO, use the following request:

POST /container_servers/:container_server_id/startup.xml
POST /container_servers/:container_server_id/startup.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where:

`container_server_id` - the ID of the container server you want to boot

`iso_id` - the ID of the ISO you want to boot from

27.23 Suspend Container Server

To suspend a container server, use the following request:

POST /container_servers/:container_server_id/suspend.xml
POST /container_servers/:container_server_id/suspend.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/23/suspend.xml
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/23/suspend.json
```

Where:

`container_server_id` - ID of a container server you want to suspend

27.24 Unlock Container Server

To unlock a container server, use the following request:
POST /container_servers/:container_server_id/unlock.xml
POST /container_servers/:container_server_id/unlock.json

**XML Request Example**

curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/23/unlock.xml

**JSON Request Example**

curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/23/unlock.json

### 27.25 Unsuspend Container Server

To activate a container server again, use the same request as to suspend it:

POST /container_servers/:container_server_id/suspend.xml
POST /container_servers/:container_server_id/suspend.json

For details, refer to the [Suspend Container Server](#) section.

### 27.26 Shut down Container Server

To shut down a container server, use the following request:

POST /container_servers/:container_server_id/shutdown.xml
POST /container_servers/:container_server_id/shutdown.json

**XML Request Example**

curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/34/shutdown.xml

**JSON Request Example**

curl -i -X POST -u user:userpass --url http://onapp.test/container_servers/34/shutdown.json

### 27.27 Stop Container Server

To stop a container server, use the following request:

POST /container_servers/:container_server_id/stop.xml
POST /container_servers/:container_server_id/stop.json

**XML Request Example**
27.28 Open Container Server Console

To open a container server console:

1. Run the following request:
   - GET /container_servers/:container_server_id/console.xml
   - GET /container_servers/:container_server_id/console.json

2. Find and copy the value for the remote_key parameter in the response output.
3. Open the following URL in the browser:
   - http://onapp.test/console_remote/[remote_key_parameter_value]

27.29 Container Server Billing Statistics

To view the billing statistics for a particular container server, use the following request:

GET /container_servers/:container_server_id/vm_stats.xml
GET /container_servers/:container_server_id/vm_stats.json

XML Request Example

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/container_servers/23/vm_stats.xml
```

JSON Request Example

```
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/container_servers/23/vm_stats.json
```
<vm_hourly_stats type="array">
<vm_hourly_stat>
<created_at type="datetime">2011-08-09T12:00:10Z</created_at>
<currency_code>USD</currency_code>
{id type="integer">8248</id>
<stat_time type="datetime">2011-08-09T12:00:10Z</stat_time>
<updated_at type="datetime">2011-08-09T12:00:10Z</updated_at>
<user_id type="integer">1</user_id>
<container_server_id type="integer">44</container_server_id>
<vm_billing_stat_id type="integer">100175</vm_billing_stat_id>
<billing_stats>
<disks type="array">
<disk>
<id type="integer">2933</id>
<cost>
<value type="integer">5</value>
<cost type="float">3.0</cost>
<resource_name>disk_size</resource_name>
</cost>
</disk>
</disks>
<network_interfaces type="array">
<network_interface>
<id type="integer">2688</id>
<cost>
<value type="integer">1</value>
<cost type="float">0.0</cost>
<resource_name>ip_addresses</resource_name>
</cost>
</network_interface>
</network_interfaces>
<container_servers type="array">
<container_server>
<id type="integer">1701</id>
<cost>
<value type="integer">1</value>
<cost type="float">0.0</cost>
<resource_name>cpus</resource_name>
</cost>
</container_server>
</container_servers>
</billing_stats>
<total_cost type="float">0.0</total_cost>
<vm_resources_cost type="float">0.0</vm_resources_cost>
<usage_cost type="float">0.0</usage_cost>
</vm_hourly_stat>
</vm_hourly_stats>

Where:

created_at – the timestamp in DB when this record was created

currency_code - currency in which this container server is charged within the bucket
id – the ID of the server hourly statistics. You can add this parameter to the request URL to get a shorter statistics output.

stat_time – the particular hour for which these statistics were generated

updated_at – the date when these statistics were updated

user_id - the ID of container server owner

container_server_id - the ID of the container server

vm_billing_stat_id - billing statistics ID

billing_stats - an array of billing details for the resources used by this container server

When generating billing statistics, OnApp takes the last state of the container server during the hour. For example, if a container server was turned on at 6.15 and turned off at 6.59 it will be considered as being off for the whole hour and its resources will be billed according to the OFF prices set in the bucket. However, the container server's disk and network interface usage can still be billed in case the container server was on during that hour.

disks - an array of disks used by this container server with their billing details:

   id - disk ID used in database

   costs - an array of disk related resources with their total prices for the period specified in the stat-time parameter, where:

   value - the amount of resources used (GBs of disk size, Kbs of data read/written, the number of reads/writes)

   cost - the total due for the resource

   resource_name - the resource in question. This can be disk_size, data_read, data_written, reads_completed and writes_completed

   label - disk name used in UI

network_interfaces - an array of network interfaces used by this container server with their billing statistics:

   id - network interface ID

   costs - an array of network interface related resources with their total prices for the period specified in the stat-time parameter, where:

   value - the amount of resources used by this network interface (the number of IPs, the port speed in Mb per second, the data sent and received in Gbs )

   cost - the total due for the resource

   resource_name - the resource in question. This can be ip_addresses, rate, data_received and data_sent

   label - network interface name used in OnApp

container_server - an array of container server billing details:

   id - container server ID

   costs - an array of container server resources with their total prices for the period specified in the stat-time parameter, where:

   value - the amount of resources allocated to this container server. For the templates resource, this parameter means a template ID in database.
**cost** - the total due for this resource

**resource_name** - the resource in question. This can be cpu_shares, cpus, memory, cpu_usage and template

**label** - container server name

**total_cost** – the total amount of money owed for the container server specified by id parameter for a particular hour specified by stat_time parameter (total_cost = vm_resources_cost + usage_cost)

**vm_resources_cost** – the amount of money due for the container server resources for the particular hour specified by stat_time parameter (memory, disks, templates)

**usage_cost** – the total due for container server usage for this particular hour specified by stat_time parameter (data sent/received, bandwidth, CPU usage)

### 27.30 Search Container Server by Label

To search container servers by label, use the following request:

GET /container_servers.xml?q=label
GET /container_servers.json?q=label

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where you have to specify the label of a container server you are searching for.

### 27.31 Get Container Server CPU Usage Statistics

To view CPU usage statistics of a container server, use the following request:

GET /container_servers/:container_server_id/cpu_usage.xml
GET /container_servers/:container_server_id/cpu_usage.json

**XML Request Example**

```bash
curl -i -X GET -u user:userpass --url http://onapp.test/container_servers/223/cpu_usage.xml
```

**XML Request Example**
curl -i -X GET -u user:userpass --url http://onapp.test/container_servers/223/cpu_usage.json

Where you have to specify the container server ID.

**XML Output Example**

```xml
<cpu_hourly_stats type="array">
  <cpu_hourly_stat>
    <cpu_time type="integer">18</cpu_time>
    <created_at type="datetime">2015-01-06T10:00:18Z</created_at>
    <id type="integer">935848</id>
    <stat_time type="datetime">2015-01-06T10:00:00Z</stat_time>
    <updated_at type="datetime">2015-01-06T10:18Z</updated_at>
    <user_id type="integer">1</user_id>
    <container_server_id type="integer">1701</container_server_id>
  </cpu_hourly_stat>
  ...
</cpu_hourly_stats>
```

Where:

- **cpu_time** - use the following formula to convert CPU data received in the API output:
  
  \[ \text{CPU} = \frac{\text{cpu_time}}{10/3600} \]
  
  Where cpu_time is data from API output.
  
  For example: cpu_time = 2330, then: 2330/10/3600=0.06 (6%).
  
  We use “cpu_time” * 10 to correct store fractional values.

- **created_at** - the timestamp in DB when this record was created
- **id** - the statistics ID
- **stat_time** - the particular hour for which these statistics were generated
- **updated_at** - the time stamp in DB when this record was updated
- **user_id** - the ID of the container server owner
- **container_server_id** - ID of the container server

### 27.32 Add/Edit Admin/User Note for Container Server

To edit/make an admin note, use the following request:

PUT /container_servers/:container_server_id.xml

PUT /container_servers/:container_server_id.json

or

PUT /container_servers/:container_server_id/admin_note.xml

PUT /container_servers/:container_server_id/admin_note.json

**XML Request Example**
27.32.1 Add/Edit User Note

To edit/make a user note, use the following request:

PUT /container_servers/:container_server_id.xml
PUT /container_servers/:container_server_id.json

or

PUT /container_servers/:container_server_id/note.xml
PUT /container_servers/:container_server_id/note.json

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/container_servers/13.xml
-d '<container_server>
    <admin_note>agfagwe tiuuytjgh yuytu</admin_note>
</container_server>' -H 'Accept:application/xml' -H 'Content-type:application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/container_servers/13.json
-d '{"container_server":{"admin_note":"kjfjhjtrtjt"}}' -H 'Accept:application/json' -H 'Content-type:application/json'
```

Where:

- **admin_note** – enter the text of your note.
```bash
curl -i -X PUT -u user:userpass
http://onapp.test/container_servers/13/note.xml -d
  '<container_server><note>agfagwe tiuuytjgh
  yuytu</note></container_server>'
  -H 'Accept:application/xml' -H 'Content-type:application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT -u user:userpass
http://onapp.test/container_servers/13.json -d
  '{"container_server":{"note":"kjfhjtrtjt"}}'
  -H 'Accept:application/json'
  -H 'Content-type:application/json'
```

or

```bash
curl -i -X PUT -u user:userpass
http://onapp.test/container_servers/13/note.json -d
  '{"container_server":{"note":"kjfhjtrtjt"}}'
  -H 'Accept:application/json'
  -H 'Content-type:application/json'
```

**Where:**

*note* – enter the text of your note.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no container server with a requested ID, or URL is incorrect.
28 Control Panel Maintenance

Control panel maintenance is a tool which enables administrators to block the CP. Administrators having permissions on managing Sysadmin Tools will have access to the Control Panel as usual. However, the CP will be blocked for all other users. Servers and services will remain running.

28.1 Get Control Panel Maintenance Status

To view the status of Control Panel maintenance, use the following request:

GET /sysadmin_tools/maintenance_mode.xml
GET /sysadmin_tools/maintenance_mode.json

XML Request Example

curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass 
http://onapp.test/sysadmin_tools/maintenance_mode.xml

JSON Request Example

curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass 
http://onapp.test/sysadmin_tools/maintenance_mode.json

XML Output Example

<maintenance-mode>
  <status type="symbol">disabled</status>
</maintenance-mode>

Where:
status - the status of Control Panel maintenance. Can be "enabled" or "disabled".

28.2 Enable Control Panel Maintenance

To enable maintenance for the Control Panel, use the following request:

PUT /sysadmin_tools/maintenance_mode/enable.xml
PUT /sysadmin_tools/maintenance_mode/enable.json

XML Request Example

curl -i -X PUT -user:userpass -H 'Accept: application/xml' 
http://onapp.test/sysadmin_tools/maintenance_mode/enable.xml 'Content-type: application/xml'

JSON Request Example


After you enable maintenance for the Control Panel, all API requests will return the 503 error, if issued by users who do not have the 'Any actions on sysadmin tools' permission enabled. Users who have this permission will be able to operate the CP as usual.

### 28.3 Disable Control Panel Maintenance

To disable maintenance for the Control Panel, use the following request:

```plaintext
PUT /sysadmin_tools/maintenance_mode/disable.xml
PUT /sysadmin_tools/maintenance_mode/disable.json
```

**XML Request Example**

```bash
curl -i -X PUT -user:userpass -H 'Accept: application/xml'
http://onapp.test/sysadmin_tools/maintenance_mode/disable.xml 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT -user:userpass -H 'Accept: application/json'
http://onapp.test/sysadmin_tools/maintenance_mode/disable.json 'Content-type: application/json'
```
29 Currencies

This class allows you to set up the currency for your payments. There are four currencies in a default installation: USD, EUR, GBP and JPY. You can add more currencies at any time.

29.1 Get List of Currencies

To get the list of available currencies, use the following request:

GET /settings/currencies.xml
GET /settings/currencies.json

XML Request Example


JSON Request Example


XML Output Example

<currencies>
  <currency>
    <name>United States dollar</name>
    <format>%u%n</format>
    <created_at>2011-03-02T12:09:36+02:00</created_at>
    <code>USD</code>
    <updated_at>2011-03-22T16:12:41+02:00</updated_at>
    <id>1</id>
    <unit>$</unit>
    <separator>.</separator>
    <precision>5</precision>
    <precision_for_unit>2</precision_for_unit>
    <delimiter>,</delimiter>
  </currency>
</currencies>

Where:

name – the currency label

format - how the currency is displayed in the control panel. The following parameters are used: %n (for the digits), %u (for the currency symbol)

created_at – the date when the record in DB was added

updated_at – the date when the record in DB was updated

code - three-character currency code that is generally used to represent the currency

id – the ID of the currency

unit – a currency symbol
**separater** - a character used to format decimal numbers, e.g. 100.99

**precision** - the number of digits after the delimiter. This parameter is used when showing the costs total for a certain period, e.g. Outstanding amount, Total Cost, Payments.

**precision_per_unit** - the number of digits after the delimiter. The precision per unit parameter is used to display the prices for the resources, e.g. for CPU, Disk size, RAM, IP, Data stores, Edge servers, Disks, Backups, Templates, etc.

**delimiter** - a grouping character used to separate thousands, e.g. 100,000,000.

### 29.2 Get Currency Details

To get details for a particular currency, use the following request:

```
GET /settings/currencies/:id.xml
GET /settings/currencies/:id.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<currency>
  <name>British pound</name>
  <format>%n</format>
  <created_at>2011-03-02T12:09:36+02:00</created_at>
  <code>GBP</code>
  <updated_at>2011-03-22T15:31:10+02:00</updated_at>
  <id>2</id>
  <unit>f</unit>
  <separator>,</separator>
  <precision>1</precision>
  <precision_for_unit>2</precision_for_unit>
  <delimiter>£</delimiter>
</currency>
```

Where:

- **name** – the currency label
- **format** - how the currency is displayed in the control panel. The following parameters are used: %n (for the digits), %u (for the currency symbol)
- **created_at** – the date when the record in DB was added
- **updated_at** – the date when the record in DB was updated
- **code** - three-character currency code that is generally used to represent the currency
- **id** – the ID of the currency
unit – a currency symbol

separator - a character used to format decimal numbers, e.g 100.99

precision - the number of digits after the delimiter. This parameter is used when showing the costs total for a certain period, e.g. Outstanding amount, Total Cost, Payments.

precision_per_unit - the number of digits after the delimiter. The precision per unit parameter is used to display the prices for the resources, e.g. for CPU, Disk size, RAM, IP, Data stores, Edge servers, Disks, Backups, Templates, etc.

delimiter - a grouping character used to separate thousands, e.g: 100,000,000.

29.3 Add Currency

To add a currency, use the following request:

POST /settings/currencies.xml
POST /settings/currencies.json

XML Request Example

```
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<currency><name>Ukrainian Грувна</name><unit>§</unit><format>%n%u</format><code>UAH</code><separator>.</separator><precision>2</precision><precision_for_unit>4</precision_for_unit><delimiter>,</delimiter></currency>' --url http://onapp.test/settings/currencies.xml
```

JSON Request Example

```
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"currency":{"name":"Polski Zloti","unit":"zł","format":"%n%u","code":"POZ","separator":",","precision":"4","precision_for_unit":"4","delimiter":""}}' --url http://onapp.test/settings/currencies.json
```

Where:

delimiter - a grouping character used to separate thousands, e.g: 100,000,000.

- Be aware, that it is prohibited to set the delimiter and separator which are identical.
- The precision cannot exceed 8 symbols.
29.4 Edit Currency

To edit details of a currency, use the following request:

**PUT /settings/currencies/:id.xml**

**PUT /settings/currencies/:id.json**

**XML Request Example**

```
curl -i -X PUT -u user:secret -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '
  <currency>
    <name>British_changed</name>
    <unit>§</unit>
    <format>%n%u</format>
    <separator>.</separator>
    <precision>2</precision>
  </currency>' --url http://onapp.test/settings/currencies/5.xml
```

**JSON Request Example**

```
curl -i -X PUT -u user:secret -H 'Accept: application/json' -H 'Content-type: application/json' -d '
  {"currency":{"name":"British_changed","unit":"§","format":"%n%u","separator":"."","precision":2,"precision_for_unit":4,"delimiter":""}}' --url http://onapp.test/settings/currencies/5.json
```

Where:

- **name** – the currency label
- **unit** – a currency symbol (€, £, etc.)
- **format** - how the currency is displayed in the control panel. The following parameters are used:
  - %n (for the digits), %u (for the currency symbol)
- **code** - three-character currency code that is generally used to represent the currency
- **separator** - a character used to format decimal numbers, e.g. 100.99
- **precision** - the number of digits after the delimiter to display the costs
- **precision_for_unit** – the number of digits after the delimiter to display the prices for resources
- **delimiter** - a grouping character used to separate thousands, e.g: 100,000,000.

- Be aware, that it is prohibited to set the delimiter and separator which are identical.
- The precision cannot exceed 8 symbols.

29.5 Delete Currency

To delete a currency, use the following request:

**DELETE /settings/currencies/:id.xml**

**DELETE /settings/currencies/:id.json**
XML Request Example

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/currencies/6.xml
```

JSON Request Example

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/currencies/6.json
```

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no currency with a requested ID, or URL is incorrect.
30 Custom Recipe Variables

Custom variables are name-value sets that can be used in server and template recipes. You can create custom variables for virtual, smart and baremetal servers and image templates. All API calls are available to this class.

30.1 Get List of Custom Variables

To get the list of virtual server custom variables, use the following request:

GET /virtual_machines/:virtual_machine_id/custom_recipe_variables.xml
GET /virtual_machines/:virtual_machine_id/custom_recipe_variables.json

XML Request Example:


JSON Response Example:


Where you have to specify ID of a virtual server in the URL.

XML Output Example

<custom_recipe_variables type="array">
<custom_recipe_variable>
  <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">2</id>
  <name>sample</name>
  <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
  <value>1234</value>
  <virtual_machine_id type="integer">3898</virtual_machine_id>
</custom_recipe_variable>
</custom_recipe_variables>

Where:

custom_recipe_variable - an array of custom variable details:
created_at - the time when the variable was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at - the time when the variable was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
enabled - true, if the variable is enabled, otherwise false
id - variable ID
name - variable name
value - variable value script
virtual_machine_id - ID of a virtual machine the variable belongs to

30.2 Get Custom Variable Details

To get the list of virtual server custom variables, use the following request:

GET /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.xml
GET /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.json

XML Request Example:

```
```

JSON Response Example:

```
```

Where you have to specify ID of a virtual server in the URL.

XML Output Example

```
<custom_recipe_variable>
    <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
    <enabled type="boolean">true</enabled>
    <id type="integer">2</id>
    <name>sample</name>
    <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
    <value>value_sample</value>
    <virtual_machine_id type="integer">3898</virtual_machine_id>
</custom_recipe_variable>
```

Where:

custom_recipe_variable - an array of custom variable details:

created_at - the time when the variable was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
updated_at - the time when the variable was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
enabled - true, if the variable is enabled, otherwise false
id - variable ID
name - variable name
value - variable value script
virtual_machine_id - ID of a virtual machine the variable belongs to
30.3 Edit Custom Variable

To edit a virtual server custom recipe variable, use the following request:

PUT
/virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.xml
PUT
/virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.json

**XML Request Example:**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<custom_recipe_variable><name>varname</name><value>value</value><enabled>1</enabled></custom_recipe_variable>' --url
  http://onapp.test/virtual_machines/3898/custom_recipe_variables/2.xml
```

**JSON Response Example:**

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"custom_recipe_variable":{"name":"varname", "value":"varvalue", "enabled":"1"}}' --url
  http://onapp.test/virtual_machines/3898/custom_recipe_variables/2.json
```

**Where:**

- **enabled** - true, if the variable is enabled, otherwise false
- **name** - variable name
- **value** - variable value script

Returns HTTP 204 response on success, or HTTP 404 when a variable with the ID specified is not found.

30.4 Add Custom Variable

To create a virtual server custom variables, use the following request:

POST /virtual_machines/:virtual_machine_id/custom_recipe_variables.xml
POST /virtual_machines/:virtual_machine_id/custom_recipe_variables.json

**XML Request Example:**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<custom_recipe_variable><name>varname</name><value>value</value><enabled>1</enabled></custom_recipe_variable>' --url
  http://onapp.test/virtual_machines/3898/custom_recipe_variables.xml
```

**JSON Response Example:**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"custom_recipe_variable":{"name":"varname", "value":"varvalue", "enabled":"1"}}' --url
  http://onapp.test/virtual_machines/3898/custom_recipe_variables.xml
```

Where:

- **enabled** - true, if the variable is enabled, otherwise false
- **id** - variable ID
- **name** - variable name
- **value** - variable value script

You also have to specify ID of a virtual server in the URL.

**XML Output Example**

```xml
<custom_recipe_variable>
    <created_at type="datetime">2013-05-27T10:15:54+03:00</created_at>
    <enabled type="boolean">true</enabled>
    <id type="integer">7</id>
    <name>varname</name>
    <updated_at type="datetime">2013-05-27T10:15:54+03:00</updated_at>
    <value>varvalue</value>
    <virtual_machine_id type="integer">3992</virtual_machine_id>
</custom_recipe_variable>
```

### 30.5 Delete Custom Variable

To delete a custom variable, use the following request:

DELETE /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.xml

DELETE /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.json

**XML Request Example:**

```bash
```

**JSON Response Example:**

```bash
```
Returns HTTP 204 response on successful deletion, or HTTP 404 when a variable with the ID specified is not found.

30.6 Get List of Virtual Server Custom Variables

To get the list of virtual server custom variables, use the following request:

GET /virtual_machines/:virtual_machine_id/custom_recipe_variables.xml
GET /virtual_machines/:virtual_machine_id/custom_recipe_variables.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where you have to specify ID of a virtual server in the URL.

**XML Output Example**

```
<custom_recipe_variables type="array">
    <custom_recipe_variable>
        <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
        <enabled type="boolean">true</enabled>
        <id type="integer">2</id>
        <name>sample</name>
        <parent_id type="integer">1476</parent_id>
        <parent_type>VirtualMachine</parent_type>
        <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
        <value>1234</value>
    </custom_recipe_variable>
</custom_recipe_variables>
```

Where:

- `custom_recipe_variable` - an array of custom variable details:
  - `created_at` - the time when the variable was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - `updated_at` - the time when the variable was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - `enabled` - true, if the variable is enabled, otherwise false
  - `id` - variable ID
  - `name` - variable name
- **value** - variable value script
- **parent_id** - ID of a baremetal server the variable belongs to
- **parent_type** - type of the server the variable is assigned to. Parent type is always equal to virtual machine

### 30.7 Get List of Smart Server Custom Variables

To get the list of smart server custom variables, use the following request:

GET /smart_servers/:smart_server_id/custom_recipe_variables.xml
GET /smart_servers/:smart_server_id/custom_recipe_variables.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where you have to specify ID of a smart server in the URL.

**XML Output Example**

```xml
<custom_recipe_variables type="array">
  <custom_recipe_variable>
    <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
    <enabled type="boolean">true</enabled>
    <id type="integer">2</id>
    <name>sample</name>
    <parent_id type="integer">1476</parent_id>
    <parent_type>VirtualMachine</parent_type>
    <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
    <value>1234</value>
  </custom_recipe_variable>
</custom_recipe_variables>
```

Where:

- **custom_recipe_variable** - an array of custom variable details:
  - **created_at** - the time when the variable was created in the [YYYY][MM][DD][hh][mm][ss]Z format
  - **updated_at** - the time when the variable was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
  - **enabled** - true, if the variable is enabled, otherwise false
  - **id** - variable ID
  - **name** - variable name
• `value` - variable value script
• `parent_id` - ID of a smart server the variable belongs to
• `parent_type` - type of the server the variable is assigned to. Parent type is always equal to virtual machine

### 30.8 Get List of Baremetal Server Variables

To get the list of baremetal server custom variables, use the following request:

**GET**
/baremetal_servers/:baremetal_server_id/custom_recipe_variables.xml
**GET**
/baremetal_servers/:baremetal_server_id/custom_recipe_variables.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where you have to specify ID of a baremetal server in the URL.

**XML Output Example**

```xml
<custom_recipe_variables type="array">
<custom_recipe_variable>
  <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">2</id>
  <name>sample</name>
  <parent_id type="integer">1476</parent_id>
  <parent_type>VirtualMachine</parent_type>
  <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
  <value>1234</value>
</custom_recipe_variable>
</custom_recipe_variables>
```

**Where:**

- `custom_recipe_variable` - an array of custom variable details:
  - `created_at` - the time when the variable was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - `updated_at` - the time when the variable was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - `enabled` - true, if the variable is enabled, otherwise false
  - `id` - variable ID
### 30.9 Get Virtual Server Custom Variable Details

To get the list of virtual server custom variables, use the following request:

GET /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.xml

GET /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where you have to specify ID of a virtual server in the URL.

**XML Output Example**

```
<custom_recipe_variable>
  <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">2</id>
  <name>sample</name>
  <parent_id type="integer">1476</parent_id>
  <parent_type>VirtualMachine</parent_type>
  <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
  <value>value_sample</value>
</custom_recipe_variable>
```

Where:

- **custom_recipe_variable** - an array of custom variable details:
  - **created_at** - the time when the variable was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - **updated_at** - the time when the variable was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - **enabled** - true, if the variable is enabled, otherwise false
id - variable ID
name - variable name
value - variable value script
virtual_machine_id - ID of a virtual machine the variable belongs to

30.10 Get Smart Server Custom Variable Details

To get the list of smart server custom variables, use the following request:

GET /smart_servers/:smart_server_id/custom_recipe_variables/:custom_recipe_variable_id.xml
GET /smart_servers/:smart_server_id/custom_recipe_variables/:custom_recipe_variable_id.json

XML Request Example


JSON Request Example


Where you have to specify ID of a smart server in the URL.

XML Output Example

<custom_recipe_variable>
  <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">2</id>
  <name>sample</name>
  <parent_id type="integer">1476</parent_id>
  <parent_type>VirtualMachine</parent_type>
  <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
  <value>value_sample</value>
</custom_recipe_variable>

Where:
custom_recipe_variable - an array of custom variable details:
created_at - the time when the variable was created in the [YYYY][MM][DD]T[hh][mm][ss]Z format
updated_at - the time when the variable was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format
enabled - true, if the variable is enabled, otherwise false
id - variable ID
name - variable name
value - variable value script
parent_id - ID of a smart server the variable belongs to
parent_type - type of the server the variable is assigned to. Parent type is always equal to virtual machine

30.11 Get Baremetal Server Custom Variable Details

To get the list of baremetal server custom variables, use the following request:

GET
/baremetal_servers/:baremetal_server_id/custom_recipe_variables/:custom_recipe_variable_id.xml
GET
/baremetal_servers/:baremetal_server_id/custom_recipe_variables/:custom_recipe_variable_id.json

XML Request Example

```
```

JSON Request Example

```
```

Where you have to specify ID of a baremetal server in the URL.

XML Output Example

```
<custom_recipe_variable>
  <created_at type="datetime">2013-05-24T11:56:25+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">2</id>
  <name>sample</name>
  <parent_id type="integer">1476</parent_id>
  <parent_type>VirtualMachine</parent_type>
  <updated_at type="datetime">2013-05-24T11:56:25+03:00</updated_at>
  <value>value_sample</value>
</custom_recipe_variable>
```

Where:

custom_recipe_variable - an array of custom variable details:
created_at - the time when the variable was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at - the time when the variable was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
enabled - true, if the variable is enabled, otherwise false
id - variable ID
**name** - variable name

**value** - variable value script

**parent_id** - ID of a baremetal server the variable belongs to

**parent_type** - type of the server the variable is assigned to. Parent type is always equal to virtual machine

### 30.12 Add Virtual Server Custom Variable

To create a virtual server custom variable, use the following request:

```plaintext
POST /virtual_machines/:virtual_machine_id/custom_recipe_variables.xml
POST /virtual_machines/:virtual_machine_id/custom_recipe_variables/json
```

#### XML Request Example

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<custom_recipe_variable><name>varname</name><value>value</value><enabled>1</enabled></custom_recipe_variable>' --url http://onapp.test/virtual_machines/3898/custom_recipe_variables.xml
```

#### JSON Request Example

```bash
```

Where:

**enabled** - true, if the variable is enabled, otherwise false

**id** - variable ID

**name** - variable name

**value** - variable value script

You also have to specify ID of a virtual server in the URL.

#### XML Output Example

```xml
<custom_recipe_variable>
  <created_at type="datetime">2013-05-27T10:15:54+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">7</id>
  <name>varname</name>
  <updated_at type="datetime">2013-05-27T10:15:54+03:00</updated_at>
  <value>varvalue</value>
  <parent_id type="integer">1476</parent_id>
  <parent_type>VirtualMachine</parent_type>
</custom_recipe_variable>
```
30.13 Add Smart Server Custom Variable

To create a smart server custom variable, use the following request:

POST /smart_servers/:smart_server_id/custom_recipe_variables.xml
POST /smart_servers/:smart_server_id/custom_recipe_variables/json

XML Request Example

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<custom_recipe_variable><name>varname</name><value>value</value><enabled>1</enabled></custom_recipe_variable>' --url
  http://onapp.test/smart_servers/3898/custom_recipe_variables.xml
```

JSON Request Example

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"custom_recipe_variable":{"name":"varname", "value":"varvalue", "enabled":"1"}}' --url
  http://onapp.test/smart_servers/3898/custom_recipe_variables.json
```

Where:

- **enabled** - true, if the variable is enabled, otherwise false
- **id** - variable ID
- **name** - variable name
- **value** - variable value script

You also have to specify ID of a smart server in the URL.

XML Output Example

```xml
<custom_recipe_variable>
  <created_at type="datetime">2013-05-27T10:15:54+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">7</id>
  <name>varname</name>
  <updated_at type="datetime">2013-05-27T10:15:54+03:00</updated_at>
  <value>varvalue</value>
  <parent_id type="integer">1476</parent_id>
  <parent_type>VirtualMachine</parent_type>
</custom_recipe_variable>
```
30.14 Add Baremetal Server Custom Variable

To create a baremetal server custom variable, use the following request:

POST /baremetal_servers/:baremetal_server_id/custom_recipe_variables.xml
POST /baremetal_servers/:baremetal_server_id/custom_recipe_variables/json

XML Request Example

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '
  <custom_recipe_variable><name>varname</name><value>value</value><enabled>1</enabled></custom_recipe_variable>
' --url http://onapp.test/baremetal_servers/3898/custom_recipe_variables.xml
```

JSON Response Example

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{
  "custom_recipe_variable":{
    "name":"varname",
    "value":"varvalue",
    "enabled":"1"
  }
}' --url http://onapp.test/baremetal_servers/3898/custom_recipe_variables.json
```

Where:

- **enabled** - true, if the variable is enabled, otherwise false
- **id** - variable ID
- **name** - variable name
- **value** - variable value script

You also have to specify ID of a baremetal server in the URL.

XML Output Example

```
<custom_recipe_variable>
  <created_at type="datetime">2013-05-27T10:15:54+03:00</created_at>
  <enabled type="boolean">true</enabled>
  <id type="integer">7</id>
  <name>varname</name>
  <updated_at type="datetime">2013-05-27T10:15:54+03:00</updated_at>
  <value>varvalue</value>
  <parent_id type="integer">1476</parent_id>
  <parent_type>VirtualMachine</parent_type>
</custom_recipe_variable>
```

30.15 Edit Virtual Server Custom Variable

To edit a virtual server custom recipe variable, use the following request:
PUT /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.xml
PUT /virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.json

XML Request Example

<table>
<thead>
<tr>
<th>curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '&lt;custom_recipe_variable&gt;&lt;name&gt;varname&lt;/name&gt;&lt;value&gt;value&lt;/value&gt;&lt;enabled&gt;1&lt;/enabled&gt;&lt;/custom_recipe_variable&gt;' --url <a href="http://onapp.test/virtual_machines/3898/custom_recipe_variables/2.xml">http://onapp.test/virtual_machines/3898/custom_recipe_variables/2.xml</a></th>
</tr>
</thead>
</table>

JSON Request Example

|---|

Where:

- enabled - true, if the variable is enabled, otherwise false
- name - variable name
- value - variable value script

Returns HTTP 204 response on success, or HTTP 404 when a variable with the ID specified is not found.

30.16 Edit Smart Server Custom Variable

To edit a smart server custom recipe variable, use the following request:

PUT /smart_servers/:smart_server_id/custom_recipe_variables/:custom_recipe_variable_id.xml
PUT /smart_servers/:smart_server_id/custom_recipe_variables/:custom_recipe_variable_id.json

XML Request Example

<table>
<thead>
<tr>
<th>curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '&lt;custom_recipe_variable&gt;&lt;name&gt;varname&lt;/name&gt;&lt;value&gt;value&lt;/value&gt;&lt;enabled&gt;1&lt;/enabled&gt;&lt;/custom_recipe_variable&gt;' --url <a href="http://onapp.test/smart_servers/4/custom_recipe_variables/2.xml">http://onapp.test/smart_servers/4/custom_recipe_variables/2.xml</a></th>
</tr>
</thead>
</table>

JSON Request Example

|---|
OnApp Cloud 6.3 API Guide

```plaintext
```

Where:

- **enabled** - true, if the variable is enabled, otherwise false
- **name** - variable name
- **value** - variable value script

Returns HTTP 204 response on success, or HTTP 404 when a variable with the ID specified is not found.

### 30.17 Edit Baremetal Server Custom Variable

To edit a baremetal server custom recipe variable, use the following request:

**PUT**

```
/baremetal_servers/:baremetal_server_id/custom_recipe_variables/:custom_recipe_variable_id.xml
```

**PUT**

```
/baremetal_servers/:baremetal_server_id/custom_recipe_variables/:custom_recipe_variable_id.json
```

**XML Request Example**

```plaintext
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<custom_recipe_variable><name>varname</name><value>value</value><enabled>1</enabled></custom_recipe_variable>' --url http://onapp.test/baremetal_servers/4/custom_recipe_variables/2.xml
```

**JSON Request Example**

```plaintext
```

Where:

- **enabled** - true, if the variable is enabled, otherwise false
- **name** - variable name
- **value** - variable value script

Returns HTTP 204 response on success, or HTTP 404 when a variable with the ID specified is not found.
30.18 Delete Virtual Server Custom Variable

To delete a virtual server custom variable, use the following request:

DELETE
/virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.xml
DELETE
/virtual_machines/:virtual_machine_id/custom_recipe_variables/:custom_recipe_variable_id.json

XML Request Example


JSON Request Example


Returns HTTP 204 response on successful deletion, or HTTP 404 when a variable with the ID specified is not found.

30.19 Delete Smart Server Custom Variable

To delete a smart server custom variable, use the following request:

DELETE
/smart_servers/:smart_server_id/custom_recipe_variables/:custom_recipe_variable_id.xml
DELETE
/smart_servers/:smart_server_id/custom_recipe_variables/:custom_recipe_variable_id.json

XML Request Example


JSON Request Example


Returns HTTP 204 response on successful deletion, or HTTP 404 when a variable with the ID specified is not found.
To delete a baremetal server custom variable, use the following request:

DELETE
/baremetal_servers/:baremetal_server_id/custom_recipe_variables/:custom_recipe_variable_id.xml
DELETE
/baremetal_servers/:baremetal_server_id/custom_recipe_variables/:custom_recipe_variable_id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Returns HTTP 204 response on successful deletion, or HTTP 404 when a variable with the ID specified is not found.
31 Data Stores

Data stores provide disk space for your virtual servers and operating systems. Data stores are attached to compute resources, and may also form part of a data store zone. All CRUD operations are available to data stores.

There are three types of data stores supported by OnApp:
1. Traditional logical volume data stores based on a centralized SAN.
2. ESXi datastores used under VMware.
3. Integrated storage data stores (see Integrated Storage chapter for details).

31.1 Get List of Data Stores

To get the list of data stores, use the following request:
GET /settings/data_stores.xml
GET /settings/data_stores.json

XML Request Example


JSON Request Example


XML Output Example

```
<data-stores type="array">
  <data_store>
    <id type="integer">33</id>
    <label>test</label>
    <identifier>onapp-rcvbozgnipdixt</identifier>
    <created_at type="dateTime">2019-04-03T15:32:14+03:00</created_at>
    <updated_at type="dateTime">2019-04-03T15:32:21+03:00</updated_at>
    <local_hypervisor_id type="integer">5</local_hypervisor_id>
    <data_store_size type="integer">1000</data_store_size>
    <zombie_disks_size type="integer">0</zombie_disks_size>
    <ip>11.111.111.111</ip>
    <data_store_group_id type="integer">1</data_store_group_id>
    <enabled type="boolean">true</enabled>
    <data_store_type>lvm</data_store_type>
    <hypervisor_group_id type="integer">12</hypervisor_group_id>
    <io_limits>1</io_limits>
    <trim type="boolean">true</trim>
    <usage type="integer">40</usage>
  </data_store>
  ...
</data_stores>
```

Where:
id - the data store ID
label - the data store label
identifier - the data store identifier
created_at - the date when the data store was created in the [YYYY][MM][DD][T][hh][mm][ss][Z] format
updated_at - the date when the data store was updated in the [YYYY][MM][DD][T][hh][mm][ss][Z] format
local_hypervisor_id - the ID of the compute resource using this data store
data_store_size - the size of your data store shown in GB
zombie_disk_size - the size of zombie disks attached to this data store in GB
ip - the IP address of the data store
data_store_group_id - the ID of the data store zone to which the data store belongs
enabled - True if a data store is enabled and you can attach disks to it, otherwise, false.
data_store_type - data store type: lvm or vmware.
hypervisor_group_id - the ID of the compute resource zone associated with the data store
io_limits - an array of IO limits applied to the data store
trim - true if TRIM is enabled on the data store, otherwise, false
usage - the total disk usage on the data store in GB

Page History
v. 6.1 Edge 1
• added the trim parameter

31.2 Get Data Store Details
To get details of a particular data store, use the following request:
GET /settings/data_stores/:id.xml
GET /settings/data_stores/:id.json

XML Request Example
```
```

JSON Request Example
```
```

XML Output Example
<data_store>
  <label>SAN1</label>
  <created_at type="datetime">2010-10-28T03:18:51+07:00</created_at>
  <updated_at type="datetime">2011-07-19T21:42:07+07:00</updated_at>
  <zombie_disks_size type="integer">93</zombie_disks_size>
  <id type="integer">1</id>
  <enabled type="boolean">true</enabled>
  <data_store_group_id type="integer">1</data_store_group_id>
  <ip nil="true"></ip>
  <iscsi_ip nil="true"></iscsi_ip>
  <local_hypervisor_id nil="true"></local_hypervisor_id>
  <hypervisor_group_id nil="true"></hypervisor_group_id>
  <identifier>onapp-9yblt1m7opdptdp</identifier>
  <data_store_size type="integer">500</data_store_size>
  <data_store_type>lvm</data_store_type>
  <trim>false</trim>
  <usage type="integer">57</usage>
</data_store>

Where:

- **created_at** — the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- **data_store_size** — the size of your data store shown in GB
- **data_store_type** - data store type. Can be either lvm or vmware.
- **id** — the data store ID
- **label** — the data store label
- **local_hypervisor_id** — IDs of the compute resources using this Data Store
- **hypervisor_group_id** - the ID of the compute zone to which this data store is attached.
- **updated_at** — the date when the Data Store was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- **data_store_group_id** — the ID of a data store zone to which a particular data store is attached
- **zombie_disk_size** — the size of zombie disks attached to this data store in GB.
- **enabled** — true if a data store is enabled and you can attach disks to it, otherwise false.
- **trim** - true if TRIM is enabled on the data store, otherwise, false
- **usage** - the total disk usage on this data store in GB.

Page History

v. 6.1 Edge 1

- added the trim parameter

### 31.3 Add LVM Data Store

To add an LVM data store, use the following request:

```xml
POST /settings/data_stores.xml
POST /settings/data_stores.json
```

**XML Request Example**
curl -i -X POST http://onapp.test/settings/data_stores.xml -d
'<?xml version="1.0" encoding="UTF-8"?>
<data_store><label>DS_label</label><data_store_group_id>DS_zone_id</data_store_group_id><local_hypervisor_id>1</local_hypervisor_id><ip>DS_ip</ip><enabled>true</enabled><data_store_size>40</data_store_size><data_store_type>lvm</data_store_type></data_store>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

Where:

- **label**: the data store name
- **data_store_group_id**: the group to which this DS is assigned
- **ip**: the data store IP
- **enabled**: set true if data store is enabled, otherwise set false
- **data_store_size**: set size in GB
- **data_store_type**: set lvm data store type
- **local_hypervisor_id**: ID of a local compute resource

### 31.4 Add VMware Data Store

To add a VMware data store, use the following request:

**POST** /settings/data_stores.xml
**POST** /settings/data_stores.json

**XML Request Example**

```bash
curl -i POST -d
'<?xml version="1.0" encoding="UTF-8"?>
<data_store><label>testapi</label><ip>192.168.1.27</ip><enabled>1</enabled><data_store_size>40</data_store_size><local_hypervisor_id>1</local_hypervisor_id><data_store_group_id>84</data_store_group_id><data_store_type>vmware</data_store_type></data_store>' -u user:userpass
```

**JSON Request Example**

```bash
curl -i POST -d
'{"data_store":{"label":"test", "ip":"192.168.1.27", "enabled":"1", "data_store_size":"40", "local_hypervisor_id":null, "data_store_group_id":84, "data_store_type":"vmware"}}' -u user:userpass
```

Where:
label* - the data store name
data_store_group* - the group to which this DS is assigned
ip* - the data store IP
enabled* - set 1 if data store is enabled, otherwise set 0
data_store_size* - set size in GB
data_store_type - set VMware data store type
local_hypervisor_id - ID of a local compute resource

31.5 Add SolidFire Data Store

To add a SolidFire data store, use the following request:

POST /settings/data_stores.xml
POST /settings/data_stores.json

XML Request Example

curl -i -X POST -u user:userpass --url
"<data_store><label>API_SF_test_xml</label><ip>10.98.0.101</ip><data_store_type>solidfire</data_store_type><enabled>1</enabled><data_store_size>40</data_store_size><local_hypervisor_id></local_hypervisor_id><data_store_group_id>5</data_store_group_id><iscsi_ip>10.99.99.101</iscsi_ip><admin_attributes><username>onapp</username><password>password</password></admin_attributes><account_attributes><username>onapp</username><initiator_secret>ttttt123456790</initiator_secret><target_secret>ttttt123456780</target_secret></account_attributes></data_store>"

JSON Request Example

curl -i -X POST -u user:userpass --url

Where:
label* - the data store name
data_store_group* - the group to which this DS is assigned
ip* - the data store IP
enabled* - set 1 if data store is enabled, otherwise set 0
data_store_size* - set size in GB
data_store_type - set solidfire data store type
local_hypervisor_id - ID of a local compute resource
iscsi_ip - iSCSI IP address

admin_attributes - an array of cluster admin attributes:
- username - username for cluster authorization
- password - password for cluster authorization

account_attributes - an array of SolidFire account attributes:
- username - specify SolidFire account username
- initiator_secret - iSCSI initiator secret (optional)
- target_secret - iSCSI target secret (optional)

Examples

XML Output Example
<data_store>
  <created_at type="datetime">2012-11-16T11:47:18+00:00</created_at>
  <data_store_group_id type="integer">5</data_store_group_id>
  <data_store_type>solidfire</data_store_type>
  <enabled type="boolean">true</enabled>
  <id type="integer">12</id>
  <identifier>onapp-ndnminre4r4r9h</identifier>
  <ip>10.98.0.101</ip>
  <iscsi_ip nil="true"/>
  <label>API_SF_test_xml</label>
  <local_hypervisor_id nil="true"/>
  <updated_at type="datetime">2012-11-16T11:47:18+00:00</updated_at>
  <zombie_disks_size type="integer">0</zombie_disks_size>
  <usage type="integer">40</usage>
</data_store>

31.6 Edit LVM Data Store

To edit an LVM data store, run the following request:

PUT /settings/data_stores/:id.xml
PUT /settings/data_stores/:id.json

XML Request Example

```
curl -i -X PUT http://onapp.test/settings/data_stores/7.xml -d '...
```

JSON Request Example

```
curl -i -X PUT http://onapp.test/settings/data_stores/7.json -d ...
```

Where

* **label** - the desired label for the data store
* **data_store_group_id** - the ID of the data store zone to which this data store is attached
* **local_hypervisor_id** - the ID of the compute resource to which this data store is assigned
* **ip** - the data store IP address
* **enabled** - set true or false. Set true if the data store is enabled and you can create virtual servers on it, otherwise false
* **data_store_size** - the data store disk capacity in GB
* **data_store_type** - for LVM data stores, set lvm
  
  💡 **trim** - true if TRIM is enabled on the data store, otherwise, false

For integrated data stores only:

* **integrated_storage_cache_enabled** - set true or false. Set true if the integrated storage cache is enabled, otherwise false
* **integrated_storage_cache_settings** - indicate integrated storage cache settings, if **integrated_storage_cache_enabled parameter** is set to true
* **auto_healing** - set true or false. Set true if auto healing is enabled, otherwise false

Returns HTTP 204 response on successful deletion, or HTTP 404 when a data store with the ID specified is not found, or the URL requested is incorrect.

**Page History**

v. 6.1
- added the **trim** parameter

v. 5.3
- added the **auto_healing** parameter

**31.7 Edit SolidFire Data Store**

To edit a SolidFire data store, use the following request:

PUT /settings/data_stores.xml
PUT /settings/data_stores.json

**XML Request Example**

JSON Request Example


Where:

**label** - the data store name

**data_store_group** - the group to which this DS is assigned

**ip** - the data store IP

**enabled** - set 1 if data store is enabled, otherwise set 0

**data_store_size** - set size in GB

**data_store_type** - set solidfire data store type

**local_hypervisor_id** - ID of a local compute resource

**iscsi_ip** - iSCSI IP address

**admin_attributes** - an array of cluster admin attributes:

- **username** - username for cluster authorization
- **password** - password for cluster authentication

**account_attributes** - an array of SolidFire account attributes:

- **username** - specify SolidFire account username
- **initiator_secret** - iSCSI initiator secret (optional)
- **target_secret** - iSCSI target secret (optional)

Returns HTTP 204 response on successful deletion, or HTTP 404 when a data store with the ID specified is not found, or the URL requested is incorrect.
31.8 Edit Data Store IOPS Limits

To edit a data store IOPS limits, use the following request:

PUT /settings/data_stores/:id/io_limits.xml
PUT /settings/data_stores/:id/io_limits.json

**XML Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/data_stores/12/io_limits.xml -d '...

**JSON Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/data_stores/12/io_limits.json -d '...

Where:
- **read_iops** - set the read IOPS amount
- **write_iops** - set the write IOPS amount
- **read_throughput** - specify the read throughput (in MB/s)
- **write_throughput** - specify the write throughput (in MB/s)

31.9 Delete Data Store

To delete a data store, use the following request:

DELETE /settings/data_stores/:id.xml
DELETE /settings/data_stores/:id.json

**XML Request Example**

```bash

**JSON Request Example**

```bash

Returns HTTP 204 response on successful deletion, or HTTP 404 when a data store with the ID specified is not found, or the URL requested is incorrect.
32 Data Store Zones

This class manages all the Data store zones created in the cloud. A data store zone consists of several data stores sharing the same permissions and assigned to the same bucket. By setting up different zones, you can create different tiers of storage with different pricing and performance.

32.1 Get List of Data Store Zones

To get the list of available data store zones, use the following request:

GET /settings/data_store_zones.xml
GET /settings/data_store_zones.json

You will get an array of data store zones set up within your cloud.

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<data_store_groups type="array">
<data_store_group>
<label>DSZ_1</label>
<location_group_id type="integer">1</location_group_id>
<preconfigured_only type="boolean">false</preconfigured_only>
<provider_vdc_id type="integer">55</provider_vdc_id>
<created_at type="datetime">2011-01-11T11:15Z</created_at>
<updated_at type="datetime">2011-01-17T12:56:41Z</updated_at>
</data_store_group>
</data_store_groups>
```

Where:

- **label** - the data store zone title
- **location_group_id** - ID of a location group the data store zone is assigned to
- **preconfigured_only** - whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.
- **provider_vdc_id** - the provider resource pool ID
- **created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **updated_at** - the date when the Data store zone was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
32.2 Get Data Store Zone Details

To get the details of a data store, use the following request:

GET /settings/data_store_zones/:id.xml
GET /settings/data_store_zones/:id.json

This method returns details of a particular data store zone.

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```
<data-store-group>
  <label>DSZ_2</label>
  <location_group_id type="integer">1</location_group_id>
  <preconfigured_only type="boolean">false</preconfigured_only>
  <provider_vdc_id type="integer">55</provider_vdc_id>
  <created_at type="datetime">2011-01-11T11:55:00Z</created_at>
  <updated_at type="datetime">2011-01-17T12:56:27Z</updated_at>
  <id type="integer">8</id>
</data-store-group>
</data-store-groups>
```

**Where:**

- **label** - the data store zone title
- **location_group_id** - ID of a location group the data store zone is assigned to
- **preconfigured_only** - whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.
provider_vdc_id - the provider resource pool ID
created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
updated_at - the date when the Data store zone was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
id - the data store zone ID

Page History
v. 5.6
• added the provider_vdc_id parameter
v. 4.2
• added the preconfigured_only parameter
v. 3.1
• added the location_group_id parameter

32.3 Add Data Store Zone
To create a new data store zone, use the following request:

POST /settings/data_store_zones.xml
POST /settings/data_store_zones.json

XML Request Example

curl -i -X POST http://onapp.test/settings/data_store_zones.xml -d '<?xml version="1.0" encoding="UTF-8"?><data_store_group><label>TEST_XM</label><location_group_id>1</location_group_id><preconfigured_only>true</preconfigured_only></data_store_group>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X POST http://onapp.test/settings/data_store_zones.json -d '{"data_store_group":{"label":"TEST_JSON","location_group_id":"1","preconfigured_only":true}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

Where:
label* - is new data_store_zone title
location_group_id - ID of a location group you wish to assign the data store zone to
server_type - specify the type of this data store zone, it can be one of the following: virtual, smart, baremetal or vpc:

• LVM data store zones can have either the virtual or the smart type
• Integrated Storage data store zones can have either the virtual or the smart type
• SolidFire data store zones can have either the virtual or the smart type
• vCloud Director data store zones should have the vpc type
VMware data store zones should have the virtual type

`preconfigured_only` - specify whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSSs built by setting resources manually) creation.

### Page History

- **v. 5.3**
  - added the `server_type` parameter

- **v. 4.2**
  - added the `preconfigured_only` parameter

- **v. 3.1**
  - added the `location_group_id` parameter

### 32.4 Edit Data Store Zone

To edit a label of a particular data store zone, use the following request:

```
PUT /settings/data_store_zones/:id.xml
PUT /settings/data_store_zones/:id.json
```

**XML Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/data_store_zones/7.xml -d '<data_store_group><label>Data_Store_Name</label><location_group_id>1</location_group_id><preconfigured_only>true</preconfigured_only></data_store_group>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/data_store_zones/7.json -d '{"data_store_group":
{"label":"Data_Store_Name","location_group_id":"1","preconfigured_only":"true"}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

- `label` - data_store_zone title
- `location_group_id` - ID of a location group you wish to assign the data store zone to. You can change the already assigned location group only if there are no disks or ISOs built on data stores of current zone.
- `preconfigured_only` - specify whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSSs built by setting resources manually) creation.

### Page History

- **v. 4.2**
- added `preconfigured_only` parameter

v.3.1

- added the `location_group_id` parameter

### 32.5 Delete Data Store Zone

To delete a particular data store zone, use the following request:

```
DELETE /settings/data_store_zones/:id.xml
DELETE /settings/data_store_zones/:id.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

You will get a 204 status response on success, and 404 if there is no such a data store zone with a requested ID or you entered incorrect URL.

### 32.6 Get the List of Data Stores Attached to Data Store Zone

To get the list of all data stores attached to a data store zone, use the following request:

```
GET /settings/data_store_zones/:data_store_group_id/data_stores.xml
GET /settings/data_store_zones/:data_store_group_id/data_stores.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

On success, an array of data stores is returned.

**XML Output Example**
<data-stores type="array">
<data-store>
  <label>ds1</label>
  <created_at type="datetime">2011-01-06T10:54:30Z</created_at>
  <updated_at type="datetime">2011-02-07T12:27:32Z</updated_at>
  <data_store_group_id type="integer">5</data_store_group_id>
  <enabled type="boolean">false</enabled>
  <id type="integer">1</id>
  <zombie_disks_size type="integer">110</zombie_disks_size>
  <ip></ip>
  <local_hypervisor_id type="integer" nil="true"></local_compute_resource_id>
  <data_store_size type="integer">465</data_store_size>
  <identifier>onapp-ojgg2jk75zfzmw</identifier>
</data-store>
</data-stores>

Where:

- **label** - the name of the data store attached to this data store zone
- **created_at** - time when the DB record was created
- **updated_at** - time when the DB record was updated
- **data_store_group_id** - the ID of a data store zone to which this data store is attached
- **enabled** - true if the data store is enabled and you can create virtual servers on it, otherwise false
- **id** - the data store ID
- **zombie_disks_size** - the disk space in GB allocated to zombie disks
- **ip** - the data store IP address
- **local_hypervisor_id** - the ID of the compute resource to which this data store is assigned
- **data_store_size** - the data store disk capacity in GB
- **identifier** - the data store identifier in DB

### 32.7 Attach Data Store to Data Store Zone

To attach a data store to a data store zone, use the following request:

**POST**

/postings/data_store_zones/:data_store_zone_id/data_stores/:id/attach.

**XML Request Example**

```
curl -X POST
http://onapp.test/settings/data_store_zones/12/data_stores/2/attach.xml
-u user:userpass
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -X POST
http://onapp.test/settings/data_store_zones/12/data_stores/2/attach.json
-u user:userpass
-H 'Accept: application/json' -H 'Content-type: application/json'
```
Using this request you attach an unassigned data store (:data_store_id) to a data store zone (:data_store_zone_id).

When you add a data store to a data store zone, it inherits the zone's type. For more information refer to Zone Types.

32.8 Detach Data Store from Data Store Zone

To detach a data store from a data store zone, use the following request:

```
POST /settings/data_store_zones/:data_store_group_id/data_stores/:id/detach.xml
POST /settings/data_store_zones/:data_store_group_id/data_stores/:id/detach.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```
33 Disks

Disks provide space for virtual server data. A disk is a partition of a data store that is allocated to a specific virtual server. All CRUD operations are available for disks.

33.1 Get List of Disks

To get the list of all disks in the cloud, use the following request:

GET /settings/disks.xml
GET /settings/disks.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```xml
<disks type="array">
<disk>
  <mounted>true</mounted>
  <built type="boolean">true</built>
  <burst_bw type="integer">1000</burst_bw>
  <bus nil="true"/>
  <created_at type="datetime">2013-01-21T12:59:06+02:00</created_at>
  <data_store_id type="integer">6</data_store_id>
  <disk_size type="integer">60</disk_size>
  <disk_vm_number type="integer">0</disk_vm_number>
  <file_system type="symbol">ext3</file_system>
  <id type="integer">868</id>
  <identifier>wtqpz628vbdasx</identifier>
  <iqn nil="true"/>
  <is_swap type="boolean">false</is_swap>
  <label nil="true"/>
  <locked type="boolean">false</locked>
  <max_bw type="integer">1000</max_bw>
  <mount_point nil="true"/>
  <primary type="boolean">true</primary>
  <updated_at type="datetime">2013-01-21T13:00:18+02:00</updated_at>
  <virtual_machine_id type="integer">458</virtual_machine_id>
  <volume_id nil="true"/>
  <has_autobackups type="boolean">false</has_autobackups>
</disk>
</disks>
```

*Where:*

- `add_to_freebsd_fstab` - true, if this disk is added to the FreeBSD fstab, otherwise false
**add_to_linux_fstab** - true, if this disk is added to Linux fstab, otherwise false

**mounted** - set 'true' to mount the disk inside OS automatically, otherwise set 'false'

You can use a single **mounted** parameter, to substitute the two **add_to_linux_fstab** and **add_to_freebsd_fstab** parameters.

**built** - true if the disk is built, otherwise false

**created_at** - the date when the disk was created in the [YYYY][MM][DD][hh][mm][ss]Z format

**updated_at** - the date when the disk was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**data_store_id** - the ID of the data store this disk is located

**disk_size** - disk size in GB

**disk_vm_number** - the number of virtual servers using this disk

**file_system** - disk filesystem (ext3 or ext4)

**id** - the disk ID

**identifier** - disk identifier

**is_swap** - true if this is a swap disk, otherwise false

**label** - disk's label

**locked** - true if the disk is locked, otherwise false

**mount_point** - disk mount point.

**primary** - true if the disk is primary. Otherwise false.

**virtual_machine_id** - the ID of the virtual server using this disk

**volume_id** - data store ID

**has_autobackups** - true if the disk has automatic backups set up, otherwise false

**SolidFire** - related parameters:

**iqn** - volume ISCSI qualified name

**burst_bw** - maximum bandwidth allowed set in MB/sec

**max_bw** - maximum bandwidth allowed set in MB/sec

### 33.2 Get List of VS Disks

To get the list of disks available for a particular VS, use the following request:

GET /virtual_machines/:virtual_machine_id/disks.xml
GET /virtual_machines/:virtual_machine_id/disks.json

**XML Request Example**

```
curl -i -X GET -u user:userpass --url http://onapp.test/virtual_machines/458/disks.xml
```

**JSON Request Example**

...
curl -i -X GET -u user:userpass --url http://onapp.test/virtual_machines/458/disks.json

XML Output Example

```
<disks type="array">
  <disk>
    <mounted>true</mounted>
    <built type="boolean">true</built>
    <burst_bw type="integer">1000</burst_bw>
    <bus nil="true"/>
    <created_at type="datetime">2013-01-21T12:59:06+02:00</created_at>
    <data_store_id type="integer">6</data_store_id>
    <disk_size type="integer">60</disk_size>
    <disk_vm_number type="integer">0</disk_vm_number>
    <file_system type="symbol">ext3</file_system>
    <id type="integer">868</id>
    <identifier>wtqpz628vbdasx</identifier>
    <iqn nil="true"/>
    <is_swap type="boolean">false</is_swap>
    <label nil="true"/>
    <locked type="boolean">false</locked>
    <max_bw type="integer">1000</max_bw>
    <mount_point nil="true"/>
    <primary type="boolean">true</primary>
    <updated_at type="datetime">2013-01-21T13:00:18+02:00</updated_at>
    <virtual_machine_id type="integer">458</virtual_machine_id>
    <volume_id nil="true"/>
    <has_autobackups type="boolean">false</has_autobackups>
  </disk>
</disks>
```

Where:

- `add_to_freebsd_fstab` - true, if this disk is added to the FreeBSD fstab, otherwise false
- `add_to_linux_fstab` - true, if this disk is added to Linux fstab, otherwise false
- `mounted` - set ‘true’ to mount the disk inside OS automatically, otherwise set ‘false’

You can use a single `mounted` parameter, to substitute the two `add_to_linux_fstab` and `add_to_freebsd_fstab` parameters.

- `built` - true if the disk is built, otherwise false
- `created_at` - the date when the disk was created in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- `updated_at` - the date when the disk was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- `data_store_id` - the ID of the data store this disk is located
- `disk_size` - disk size in GB
- `disk_vm_number` - the number of virtual servers using this disk
- `file_system` - disk filesystem (ext3 or ext4)
- `id` - the disk ID
- `identifier` - disk identifier
- `is_swap` - true if this is a swap disk, otherwise false
label - disk's label

locked - true if the disk is locked, otherwise false

mount_point - disk mount point.

primary - true if the disk is primary. Otherwise false.

virtual_machine_id - the ID of the virtual server using this disk

volume_id - data store ID

has_autobackups - true if the disk has automatic backups set up, otherwise false.

SolidFire - related parameters:

iqn - volume ISCSI qualified name

burst_bw - maximum bandwidth allowed set in MB/sec

max_bw - maximum bandwidth allowed set in MB/sec

33.3 Get VS Disk Details

To view the details of a VS disk, use the following request:

GET /virtual_machines/:id/disks/:id.xml
GET /virtual_machines/:id/disks/:id.json

XML Request Example


JSON Request Example


XML Output Example
<disk>
  <add_to_freebsd_fstab nil="true"/>
  <add_to_linux_fstab nil="true"/>
  <built type="Boolean">true</built>
  <burst_bw type="integer">100</burst_bw>
  <created_at type="datetime">2015-03-31T11:15:51+00:00</created_at>
  <data_store_id type="integer">9</data_store_id>
  <disk_vm_number type="integer">1</disk_vm_number>
  <file_system nil="true"/>
  <id type="integer">4</id>
  <identifier>nd6550ds78cmo0</identifier>
  <iqn nil="true"/>
  <is_swap type="boolean">false</is_swap>
  <label>Hard disk 1</label>
  <locked type="boolean">false</locked>
  <max_bw type="integer">1000</max_bw>
  <mount_point>/mnt/onapp-disk-nd6550ds78cmo0</mount_point>
  <primary type="boolean">false</primary>
  <updated_at type="datetime">2015-03-31T11:15:51+00:00</updated_at>
  <virtual_machine_id type="integer">4</virtual_machine_id>
  <volume_id nil="true"/>
  <has_autobackups type="boolean">false</has_autobackups>
</disk>

Where:

add_to_freebsd_fstab - true, if this disk is added to the FreeBSD fstab, otherwise false
add_to_linux_fstab - true, if this disk is added to Linux fstab, otherwise false
built - true if the disk is built, otherwise false
created_at - the date when the disk was created in the [YYYY][MM][DD][hh][mm][ss]Z format
data_store_id - the ID of the data store this disk is located
disk_size - disk size in GB
disk_vm_number - the number of virtual servers using this disk
file_system - disk filesystem (ext3 or ext4)
id - the disk ID
identifier - disk identifier
is_swap - true if this is a swap disk, otherwise false
label - disk's label
locked - true if the disk is locked, otherwise false
mount_point - disk mount point.
primary - true if the disk is primary. Otherwise false.
updated_at - the date when the disk was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
virtual_machine_id - the ID of the virtual server using this disk
volume_id - data store ID
has_autobackups - true if the disk has automatic backups set up, otherwise false

SolidFire - related parameters:

iqn - volume ISCSI qualified name
burst_bw - maximum bandwidth allowed set in MB/sec
max_bw - maximum bandwidth allowed set in MB/sec

33.4 Add New Disk

To add a new disk, use the following request:

POST /virtual_machines/:virtual_machine_id/disks.xml
POST /virtual_machines/:virtual_machine_id/disks.json

**XML Request Example**

```bash
curl -i -X POST http://onapp.test/virtual_machines/458/disks.xml -d "<disk><primary>0</primary><disk_size>5</disk_size><file_system>ext4</file_system><data_store_id>5</data_store_id><mount_point>true</mount_point><hot_attach>1</hot_attach><min_iops>600</min_iops><is_swap>true</is_swap><require_format_disk>1</require_format_disk><mounted>true</mounted></disk>" -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X POST http://onapp.test/virtual_machines/458/disks.json -d '{"disk":{"primary":"0","disk_size":"2","file_system":"ext4","data_store_id":"5","label":"qwerq","require_format_disk":"1","mount_point":"/mnt/aaa","hot_attach":"1","min_iops":"600","mounted":"true"}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

data_store_id* - the ID of a data store where this disk is located
label - the disk label
primary - set 1 if the disk is primary, otherwise, set 0
disk_size* - the disk space in GB

Please note that support of secondary disks larger than 2 TB is limited. Certain operations might fail. This will be improved in future releases.

Click here to see the details of adding a disk 2 TB+

- If you add a disk larger than 2 TB, you will not be able to mount or create a backup of such disk via OnApp Control Panel. These operations may be performed only manually inside a virtual server. Also, resize will not be possible for such disks.
- You can perform the following operations with a secondary disk that is larger than 2 TB:
  - Migrate
  - Delete / Wipe
  - Edit IO limits
OnApp Cloud 6.3 API Guide

- Rebalance (for VSs with Integrated Storage feature enabled)
  - OnApp provides a RAW device available inside a VS that you may use. For example, you may create a GPT partition and format it using any available file system which can be mounted inside VS.

  - `is_swap` - set `true` if this is a swap disk, otherwise, set `false`
  - `mount_point` - a physical location in the partition used as a root filesystem
  - `hot_attach` - set 1 to enable hot attachment, otherwise, set 0. In this case, a virtual server is not stopped when a disk is added.

To run hot attach, a virtual server template should support VirtIO virtualization. The hot attach option is available only on KVM compute resources based on CentOS 6.x/7.x for virtual servers with VirtIO support.

- `min_iops` - the minimum number of IO operations per second (this is a SolidFire related parameter)
- `add_to_linux_fstab` - set `true` to add the disk to your Linux `fstab` file, otherwise, set `false`
- `add_to_freebsd_fstab` - set `true` to add the disk to your FreeBSD `fstab` file, otherwise, set `false`
- `mounted` - set `true` to mount the disk inside OS automatically, otherwise, set `false`

You can use a single `mounted` parameter to substitute the two `add_to_linux_fstab` and `add_to_freebsd_fstab` parameters.

- `require_format_disk` - set 1 to format disk, otherwise, set 0
- `file_system` - for Linux-based virtual servers, you can specify the `ext4` or `xfs` file system instead of the `ext3` default one. For Windows-based virtual servers, you cannot specify a file system other than `ntfs`. To set the `file_system` parameter, you must set the `require_format_disk` option.

### 33.5 Edit Disk

To edit a disk, use the following request:

```
PUT /settings/disks/:id.xml
PUT /settings/disks/:id.json
```

**XML Request Example**

```
```
JSON Request Example

```
curl -i -X PUT http://onapp.test/settings/disks/2.json -d '
{"disk":{"label":"for BU", "disk_size":"1", "require_format_disk":"1", "mounted":"true", "mount_point":"/mnt/disk1", "file_system":"ext4"}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **label** - the disk label
- **disk_size** - the disk space in GB
- **require_format_disk** - set 1 to format disk, otherwise, set 0
- **add_to_linux_fstab** - set 1 to add the disk to your Linux `fstab` file, otherwise, set 0
- **add_to_freebsd_fstab** - set 1 to add the disk to your FreeBSD `fstab` file, otherwise, set 0
- **mounted** - set 'true' to mount the disk inside OS automatically, otherwise, set 'false'.

You can use a single mounted parameter, to substitute the two `add_to_linux_fstab` and `add_to_freebsd_fstab` parameters.

- **mount_point** - a physical location in the partition used as a root file system
- **file_system** - for Linux templates, you can choose `ext4` file system instead of the `ext3` default one

To set the file system parameter, you must set the **require_format_disk** option.

- **min_iops** - minimum number of IO operations per second. This is a SolidFire-related parameter.

Returns HTTP 204 response on successful deletion, or HTTP 404 when a disk with the ID specified is not found, or the URL requested is incorrect.

You can also edit a disk through another URL:
```
onapp.test/virtual_machines/:virtual_machines_id/disks/:id
```

- You cannot decrease a size of the Integrated Storage data store disks.
- You cannot resize a disk that uses GUID Partition Table (GPT).
- You cannot decrease the disk size for Windows-based and FreeBSD-based virtual servers. Only the increase disk size option is available.
- You can't resize the primary disk for FreeBSD-based virtual servers.
Decreasing disk size for Linux-based virtual servers may lead to the filesystem inconsistencies. Make sure you have current backups before proceeding.

33.6 Edit Disk IO Limits

To edit the disk IO limits, use the following request:

PUT /settings/disks/:id/io_limits.xml
PUT /settings/disks/:id/io_limits.json

**XML Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/disks/2/io_limits.xml -d '
  <io_limits>
    <io_limits_override>true</io_limits_override>
    <read_iops>100</read_iops>
    <write_iops>100</write_iops>
    <read_throughput>1</read_throughput>
    <write_throughput>1</write_throughput>
  </io_limits>
' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT http://onapp.test/settings/disks/2/io_limits.json -d '
  {"io_limits":{"io_limits_override":true,"read_iops":"100", "write_iops":"100", "read_throughput":"1", "write_throughput":"1"}}'
' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- `io_limits_override` - set 'true' to override limits with the following settings. If you set 'false', the default settings will be applied.
- `read_iops` - set the read IOPS amount
- `write_iops` - set the write IOPS amount
- `read_throughput` - specify the read throughput (in MB/s)
- `write_throughput` - specify the write throughput (in MB/s)

33.7 Migrate Disks

To migrate a VS disk to another data store, use the following request:

POST /virtual_machines/:virtual_machine_identifier/disks/:disk_id/migration.xml
POST /virtual_machines/:virtual_machine_identifier/disks/:disk_id/migration.json

The following requests are still valid but soon will be deprecated:

POST /virtual_machines/:virtual_machine_id/disks/:disk_id/migrate.xml
POST /virtual_machines/:virtual_machine_id/disks/:disk_id/migrate.json

OnApp Cloud 6.3 API Guide
**XML Request Example**

```
curl -i -X POST -u user:userpass
http://onapp.test/virtual_machines/23/disks/1/migration.xml -d
'"disk_migration"{"type":"hot","data_store_id":"383","virtual_machine_identifier":"gdtcetkkxgkhuu","disk_id":"94"}"
-H 'Accept: application/xml'
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass
http://onapp.test/virtual_machines/23/disks/1/migration.json -d
'{"disk_migration":{"type":"hot","data_store_id":"383","virtual_machine_identifier":"gdtcetkkxgkhuu","disk_id":"94"}}'
-H 'Accept: application/json'
```

**Where:**

- **type** - the type of the migration. Set to *hot* if you wish to want to run a hot migration. For the cold migration, skip this parameter.
- **data_store_id** - the ID of the data store you migrate the disk to
- **virtual_machine_identifier** - the identifier of the VS
- **disk_id** - the ID of the disk

**Cold Migration Note**

Note that you can only move disks between data stores which are attached to the same compute resource or compute zone.

**Page History**

v.5.8
- added the following parameters:
  - `type`
  - `virtual_machine_identifier`
  - `disk_id`

**33.8 Delete Disk**

To remove a disk from a virtual server, use one of the following requests:

```
DELETE /virtual_machines/:virtual_machine_id/disks/:disk_id.xml
DELETE /virtual_machines/:virtual_machine_id/disks/:disk_id.json
```

Or

```
DELETE /settings/disks/:disk_id.xml?force=1&shutdown_type=graceful&required_s`
XML Request Example

```
curl -X DELETE
```

JSON Request Example

```
curl -X DELETE
```

Where:
- `shutdown_type` - type of the VS shutdown: hard, graceful or soft
- `required_startup` - set 1 to start up the VS automatically after deleting a disk, otherwise set 0

Returns HTTP 204 response on successful deletion, or HTTP 404 when a disk with the ID specified is not found, or the URL requested is incorrect.

### 33.9 View Disk IOPS

To view Input/Output statistics for your disks, use the following request:

GET `/settings/disks/:id/usage.xml`
GET `/settings/disks/:id/usage.json`

**XML Request Example**

```
curl -i -X GET
-H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/settings/disks/64/usage.xml
```

**JSON Request Example**

```
curl -i -X GET
-H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/settings/disks/64/usage.json
```

**XML Output Example**
<disk_hourly_stats type="array">
  <disk_hourly_stat>
    <disk_id type="integer">64</disk_id>
    <created_at type="datetime">2011-07-19T13:00:10Z</created_at>
    <updated_at type="datetime">2011-07-19T13:00:10Z</updated_at>
    <writes_completed type="integer">345685</writes_completed>
    <stat_time type="datetime">2011-07-19T13:00:00Z</stat_time>
    <data_written type="integer">11061920</data_written>
    <data_read type="integer">53840</data_read>
    <id type="integer">1028</id>
    <user_id type="integer">13</user_id>
    <virtual_machine_id type="integer">34</virtual_machine_id>
    <reads_completed type="integer">1684</reads_completed>
  </disk_hourly_stat>
  ...
</disk_hourly_stats>

Where:

disk_id - the ID of a disk
created_at - the timestamp in DB when the record was created
updated_at - the timestamp in DB when the record was updated
data_read – the amount of data read (kB) from this disk
data_written - the amount of data written (kB) to the disk
stat_time - the time when statistics were generated
writes_completed - the number of completed write operations performed during the hour for which this statistics has been gathered
reads_completed - the number of completed read operations performed during the hour for which this statistics has been gathered
user_id – ID of the user whose VS is using this disk
virtual_machine_id – ID of the VS using this disk

Use the following formulas to convert disk usage statistics data received in the API output:

(data_read / 1024) / 3600
(data_written / 1024) / 3600

33.10 Build Disk

To build a disk, use the following request:

POST /settings/disks/:id/build.xml
POST /settings/disks/:id/build.json

XML Request Example
33.11 Unlock Disk

To unlock a disk, use the following request:

POST /settings/disks/:disk_id/unlock.xml
POST /settings/disks/:disk_id/unlock.json

XML Request Example


JSON Request Example


33.12 Get List of Backups Available for Disk

To get the list of backups available to a particular disk, use the following request:

GET /settings/disks/:disk_id/backups.xml
GET /settings/disks/:disk_id/backups.json

An array of backups with their details is returned on success.

XML Request Example


JSON Request Example

XML Output Example

```
<backups type="array">
  <backup>
    <disk_id type="integer">112</disk_id>
    <built_at type="datetime">2011-07-27T15:19:47Z</built_at>
    <operating_system_distro>centos</operating_system_distro>
    <created_at type="datetime">2011-07-27T16:18Z</created_at>
    <template_id type="integer">1</template_id>
    <backup_type>days-autobackup</backup_type>
    <allowed_swap type="boolean">true</allowed_swap>
    <allow_resize_without_reboot type="boolean">true</allow_resize_without_reboot>
    <id type="integer">12</id>
    <backup_server_id type="integer">1</backup_server_id>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <backup_size>315552</backup_size>
    <min_disk_size type="integer">5</min_disk_size>
    <identifier>ytfgbj2drbs2d7</identifier>
    <locked type="boolean">false</locked>
    <built type="boolean">true</built>
  </backup>
  ...
</backups>
```

Where:

- **backup** – an array of backup details
- **disk_id** – ID of the disk
- **built_at** – time, when the disk was built
- **operating_system_distro** – distribution of the operating system
- **template_id** – ID of the template, used for assigned VS
- **operating_system** – OS of the virtual server, which is allocated at this disk
- **backup_type** – type of the backup (type of period: days/weeks/months/years)
- **allowed_swap** – true, if this is a swap disk; otherwise false
- **allow_resize_without_reboot** – true, if VS's CPU and RAM can be resized without reboot
- **id** – ID of the backup
- **backup_server_id** – the ID of the backup server where the backup is stored
- **allowed_hot_migrate** – true, if hot migration is allowed
- **backup_size** - size of the backup in Kilobytes
- **min_disk_size** – minimum disk size required in GB
- **identifier** – identifier in the DB
- **locked** – true, if the disk is locked
33.13 Assign Disk to VS

To assign a disk to VS, use the following request:

POST /settings/disks/:id/assign.xml
POST /settings/disks/:id/assign.json

XML Request Example

```bash
curl -i -X POST http://onapp.test/settings/disks/12/assign.xml -d
"<disk><temporary_virtual_machine_id>1</temporary_virtual_machine_id></disk>"
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

Where:

temporary_virtual_machine_id - ID of the VS, to which you assign the disk

33.14 Unassign Disk from VS

To unassign disk from VS, use the following request:

DELETE /settings/disks/:id/assign.xml
DELETE /settings/disks/:id/assign.json

XML Request Example

```bash
```

JSON Request Example

```bash
```
34 DRaaS Dashboard

The DRaaS Dashboard allows you to add and configure clouds, which is required for the setup of disaster recovery as a service.

34.1 API Credentials (DRaaS)

As part of the update, the DRaaS Dashboard tries to access the cloud using the specified credentials and insert data necessary for the cloud to activate DRaaS and access the Dashboard. Once this is complete, you will be able to activate DRaaS for your virtual machines.

34.1.1 Get Cloud API Credentials (DRaaS)

To get cloud API credentials, use the following request:

GET /clouds/:cloud_id/credentials.json

**JSON Request Example**

```
curl -v https://draas.io/api/3/clouds/1b74c76c1fb2b9f/credentials -H '
'Accept: application/json' -H 'Authorization: Bearer API_KEY'
```

Where:

- **API_KEY** - your API key

**JSON Output Example**

```
{"login":"admin@example.com","token":"0e34a5ead39367004f19245549a52e6dee
b46e8","url":"69.168.237.178"}
```

Where:

- **login** - login of the user to be managed by DRaaS Dashboard
- **token** - API token
- **url** - address of the cloud

34.1.2 Update Cloud API Credentials (DRaaS)

To update cloud API credentials, use the following request:

PATCH /clouds/:cloud_id/credentials.json

**JSON Request Example**

```
curl -v https://draas.io/api/3/clouds/3b7b98e4-4258-8866-4c76c1fb2b9f/credentials -X PATCH -d
'{"login":"admin@example.com","token":"0e34a5ead39367004f19245549a52e6dee
b46e8","url":"69.168.237.178"}' -H 'Accept: application/json' -H
'Authorization: Bearer API_KEY' -H 'Content-Type: application/json'
```

Where:

- **login** - login of the user to be managed by DRaaS Dashboard
34.2 Cloud (DRaaS)

OnApp DRaaS is a tool that replicates all the virtual server's data to a remote cloud in real-time. If anything happens to your replicated VS, you can quickly boot a functionally identical VS on the DRaaS cloud.

34.2.1 Register New Cloud (DRaaS)

This call only registers the cloud in DRaaS Dashboard and does not connect to the cloud. The newly registered cloud will have invalid_credentials status and will not allow registering any compute zones on it. To complete cloud registration, you should also provide correct API access credentials using another API call.

To register a new cloud on DRaaS Dashboard, use the following request:

POST /clouds.json

**JSON Request Example**


Where:

- **label** - cloud label
- **subdomain** - (case-insensitive, unique) - cloud subdomain. Once the cloud is registered, it may take a few hours to generate new SSL certificates. During this period, the subdomain.draas.io may not be accessible.
- **API_KEY** - your API key

**JSON Output Example**
{ "created_at": "2020-06-03T08:28:23.971Z", "id": "1b7b08e4-c836-4258-8866-4c76cfb2b9f", "label": "test1", "status": "invalid_credentials", "subdomain": "subdomain1", "updated_at": "2020-06-03T08:23.971Z", "user_id": "a1e74776-a870-404b-9bda-189bceb9abeb" }

Where:
created_at - the date when the cloud was registered  
id - the ID of the cloud
label - the cloud’s label
status - invalid_credentials (cloud API access credentials are incorrect and need to be updated); unavailable (cloud does not respond to API calls); or available
subdomain - (should be unique) cloud subdomain
updated_at - the date when the cloud was updated
user_id - the ID of the cloud owner

34.2.2 Get List of Clouds (DRaaS)

To get the list of clouds registered on DRaaS Dashboard, use the following request:

GET /clouds.json

JSON Request Example

```
```

Where:

API_KEY - your API key

JSON Output Example

```
[{
"created_at": "2020-06-03T12:14:02.906Z", "id": "e1fcd69a-e456-44a0-844c-f2885cd3b81a", "label": "test1", "status": "invalid_credentials", "subdomain": "subdomain11", "updated_at": "2020-06-03T12:14:02.906Z", "user_id": "a1e74776-a870-404b-9bda-189bceb9abeb" }, {
"created_at": "2020-06-03T08:28:23.971Z", "id": "1b7b08e4-c836-4258-8866-4c76cfb2b9f", "label": "test1", "status": "invalid_credentials", "subdomain": "subdomain1", "updated_at": "2020-06-03T08:23.971Z", "user_id": "a1e74776-a870-404b-9bda-189bceb9abeb" }]
```

Where:
created_at - the date when the cloud was registered  
id - the ID of the cloud
label - the cloud’s label
status - invalid_credentials (cloud API access credentials are incorrect and need to be updated); unavailable (cloud does not respond to API calls); or available
subdomain - (Should be unique) cloud subdomain
updated_at - the date when the cloud was updated
user_id - the ID of the cloud owner

34.2.3 Get Cloud Details (DRaaS)

To get the details of a specific cloud registered on DRaaS Dashboard, use the following request:
GET/clouds/:cloud_id.json

JSON Request Example


Where:
API_KEY - your API key

JSON Output Example

{"created_at":"2020-06-03T12:14:02.906Z","id":"e1fcd69a-e456-44a0-844c-f2885cd3b81a","label":"test1","status":"invalid_credentials","subdomain":"subdomain11","updated_at":"2020-06-03T12:14:02.906Z","user_id":"a1e74776-a870-404b-9bda-189bceb9abeb"}

Where:
created_at - the date when the cloud was registered
id - the ID of the cloud
label - the cloud's label
status - invalid_credentials (cloud API access credentials are incorrect and need to be updated); unavailable (cloud does not respond to API calls); or available
subdomain - (Should be unique) cloud subdomain
updated_at - the date when the cloud was updated
user_id - the ID of the cloud owner

34.2.4 Update Cloud (DRaaS)

To update a cloud on DRaaS Dashboard, use the following request:
PATCH /clouds/:cloud_id.json

JSON Request Example

"created_at": "2020-06-03T12:14:02.906Z", "id": "e1fcd69a-e456-44a0-844c-f2885cd3b81a", "label": "test1", "status": "invalid_credentials", "subdomain": "subdomain11", "updated_at": "2020-06-03T12:14:02.906Z", "user_id": "a1e74776-a870-404b-9bda-189bceb9abeb"
### 34.2.5 Remove Cloud from Dashboard (DRaaS)

To remove a cloud from a DRaaS Dashboard, use the following request:

```
DELETE /clouds/:cloud_id.json
```

**JSON Request Example**

```
curl -v http://draas.io/api/3/clouds/1b7b08e4-c836-4258-8866-4c761fb2b9f
-X DELETE -d '{"force":true}' -H 'Accept: application/json' -H 'Authorization: Bearer API_KEY' -H 'Content-Type: application/json'
```

**Where:**
- **cloud_id** - the ID of the cloud
- **force** - true or false (false by default); force removal of the cloud even if it has associated users or compute zone

With the **force** option enabled, removal will still fail if any of the cloud compute zones have associated virtual machines. Remove the virtual machine from the DRaaS Dashboard to proceed.

### API_KEY - your API key

### 34.3 Compute Zones (DRaaS)

To use DRaaS, you should have source and provider compute zones configured and connected. All API calls are available to this class.
34.3.1 Get List of Compute Zones (DRaaS)

You can get the list of all compute zones in DRaaS Dashboard; all compute zones for a specified cloud in DRaaS Dashboard; or all compute zones for a specified location in DRaaS Dashboard.

To get the paginated list of all public (and owned private) compute zones in DRaaS Dashboard, use the following request:

GET /compute-zones.json

**JSON Request Example**

```bash
```

Where:

*API_KEY* - your API key

To get the list of compute zones for a specified cloud in DRaaS Dashboard, use the following request:

GET /clouds/:cloud_id/compute-zones.json

**JSON Request Example**

```bash
```

Where:

*API_KEY* - your API key

To get the list of compute zones for a specified location in DRaaS Dashboard, use the following request:

GET /locations/:location_id/compute-zones.json

**JSON Request Example**

```bash
```

Where:

*API_KEY* - your API key

**JSON Output Example**

```json
[
{
"cloud_id": "68389528-8dbe-47da-ad47-2684e6bb669f",
"created_at": "2020-06-05T11:09:38.271Z",
"id": "c2d2a406-41d1-9a93-e151a68c589c",
"label": "IS-KVM7-ZONE",
"location_id": "a4726808-4a3d-4e1e-893d-da31246243f",
"provider": true,
"remote_id": "6",
"updated_at": "2020-06-05T11:09:38.271Z"
}
]
```
Where:

cloud_id - the ID of the cloud  
created_at - the date when the compute zone was created  
id - the ID of the compute zone  
label - the label of the compute zone  
location_id - the ID of the compute zone location  
provider - true, if it is a provider compute zone; otherwise, false  
remote_id - the ID of the compute zone in Control Panel  
updated_at - the date when the compute zone was updated

34.3.2 Get Compute Zone Details (DRaaS)

To get the details of a specific compute zone in DRaaS Dashboard, use the following request:

GET /compute-zones/:compute-zone_id.json

JSON Request Example

```
```

Where:

API_KEY - your API key

JSON Output Example

```
{
"cloud_id":"b669f","created_at":"2020-06-05T11:15:34.492Z","id":"cdc70","label":"KVM7-ZONE","location_id":"da31214623f","provider":true,"remote_id":"6","updated_at":"2020-06-05T11:15:34.492Z"
}
```

Where:

cloud_id - the ID the cloud the compute zone is related to  
created_at - the date when the compute zone was created  
id - the ID of the compute zone  
label - the label of the compute zone  
location_id - the ID of the compute zone's location  
provider - true, if it is a provider compute zone; otherwise, false  
remote_id - the ID of the compute zone in Control Panel  
updated_at - the date when the compute zone was updated

34.3.3 Register Compute Zone (DraaS)

To register a new compute zone in DRaaS Dashboard, use the following request:

POST /clouds/:cloud_id/compute-zones.json

JSON Request Example
**OnApp Cloud 6.3 API Guide**


**Where:**

- **provider** - set *true*, to mark the compute zone as a provider compute zone; otherwise, *false*
- **remote_id** - compute zone identifier in the cloud's Control Panel
- **API_KEY** - your API key

**JSON Output Example**

```
{
"cloud_id":"68389528-8dbe-47da-ad47-2684e6bb669f","created_at":"2020-06-05T15:34:49.22","id":"860486d0-b152-4be9-b6ea-c71c335c0d70","label":"IS-KVM7-ZONE","location_id":"a4726808-4a3d-4e1e-893d-da31246243f","provider":true,"remote_id":6,"updated_at":"2020-06-05T15:34:49.22"}
```

**Where:**

- **cloud_id** - the ID of the cloud
- **created_at** - the date when the compute zone was created
- **id** - the ID of the compute zone
- **label** - the label of the compute zone
- **location_id** - the ID of the compute zone location
- **provider** - *true*, if it is a provider compute zone; otherwise, *false*
- **remote_id** - the ID of the compute zone in Control Panel
- **updated_at** - the date when the compute zone was updated

**34.3.4 Update Compute Zone (DRaaS)**

The provider flag can be switched off only when the compute zone is not linked to any other compute zones.

To update the compute zone provider flag, use the following request:

```
PATCH /compute-zones/:compute-zone_id.json
```

**JSON Request Example**

```
```

**Where:**

- **provider** - *true*, if it is a provider compute zone; otherwise, *false*
- **API_KEY** - your API key
34.3.5 Delete Compute Zone (DRaaS)

The request will fail if the compute zone is still associated with virtual machines or if it is linked to any other compute zones as a provider.

To remove a compute zone from DRaaS Dashboard, use the following request:

DELETE /compute-zones/:compute-zone_id.json

JSON Request Example

```
```

Where:

- **API_KEY** - your API key

34.3.6 Get List of Compute Zone Links (DRaaS)

To get the list of all compute zone links, use the following request:

GET /compute-zone-links.json

JSON Request Example

```
```

To get the list of links of a specific compute zone, use the following request:

GET /compute-zones/:compute-zone_id/compute-zone-links.json

JSON Request Example

```
```

Where:

- **API_KEY** - your API key

JSON Output Example

```
[]
```

Where:

- **client_id** - the ID of the client compute zone
created_at - the date when the compute zone link was created
id - the ID of the compute zone link
provider_id - the ID of the provider compute zone
updated_at - the date when the compute zone link was updated for the last time

### 34.3.7 Get Compute Zone Link Details (DRaaS)

To get the details of a specific compute zone link, use the following request:

```
GET /compute-zone-links/:compute-zone-link_id.json
```

**JSON Request Example**

```
```

Where:

- **API_KEY** - your API key

**JSON Output Example**

```
{
  "client_id": "21806ea6-78b8-4f20-ada4-9f97486a7df4",
  "created_at": "2020-07-13T14:48:31.219Z",
  "id": "e7f3-a845-7b501e579e42",
  "provider_id": "a3386a9c-ee59-40c2-a949-4e6c7dcc2a32",
  "updated_at": "2020-07-13T14:48:31.219Z"
}
```

Where:

- **client_id** - the ID of the client compute zone
- **created_at** - the date when the compute zone link was created
- **id** - the ID of the compute zone link
- **provider_id** - the ID of the provider compute zone
- **updated_at** - the date when the compute zone link was updated for the last time

### 34.3.8 Create Compute Zone Link (DRaaS)

To create a compute zone link, use the following request:

```
POST /compute-zone-links.json
```

A compute zone can only have one compute zone link in which it is a client, while it can have unlimited compute zone links in which it is a provider.

**JSON Request Example**
Once the compute zones are linked, users running their virtual machines on the client compute zone will be able to activate disaster recovery, which will replicate the VM to shadow VM created on the provider compute zone.

### 34.3.9 Get Compute Zone Private Key (DRaaS)

To get the private key for a specific compute zone in DRaaS Dashboard, use the following request:

```bash
curl -v http://draas.io/api/3/compute-zones/:compute-zone_id/private-key.json
```

**JSON Request Example**

```bash
```

**Where:**

- `API_KEY` - your API key

**JSON Output Example**

```
"E8A09760A-48052DB3-A9573055-204B1A79-30141A70"
```

### 34.3.10 Regenerate Compute Zone Private Key (DRaaS)

To regenerate a private key for a compute zone in DRaaS Dashboard, use the following request:

```bash
curl -v http://draas.io/api/3/compute-zones/:compute-zone_id/private-key
```

**JSON Request Example**

```bash
```

**Where:**

- `client_id` - the ID of the client compute zone
- `provider_id` - the ID of the provider compute zone (you may also use the `private_key` parameter and specify the provider private key instead of the provider ID)
- `API_KEY` - your API key
OnApp Cloud 6.3 API Guide

API_KEY - your API key

JSON Output Example

"S0PY6EF0-6BU6FBF3-B4C3D6CA-234052D5-3082D8E4"

34.3.11 Delete Compute Zone Link (DRaaS)

To remove a compute zone link from DRaaS Dashboard, use the following request:

DELETE /compute-zone-links/:compute-zone-link_id.json

The request will fail if there are any virtual machines that are replicating in the direction of the link.

JSON Request Example

```
```

Where:

API_KEY - your API key

34.4 Disks (DRaaS)

Disks provide space for virtual machines data. You may use an API request to get the list of all disks in DRaaS Dashboard or get the details of a specific disk.

34.4.1 Get List of Disks (DRaaS)

To get the list of all disks of a specific virtual machine in DRaaS Dashboard, use the following request:

```
GET /virtual-machines/:virtual-machine_id/disks.json
```

JSON Request Example

```
```

Where:

API_KEY - your API key

JSON Output Example

Where:

- `created_at` - the date when the disk was created
- `id` - the ID of the disk
- `number` - the ordinal number of the disk on the list
- `primary` - true, if this is a primary disk; otherwise, false
- `progress` - disk replication readiness (in %)
- `replication_status` - inactive, replication_in_progress, or in sync
- `size` - the disk size (GB)
- `swap` - true, if this is a swap disk; otherwise, false
- `updated_at` - the date when the disk was updated for the last time
- `virtual_machine_id` - the disk of the virtual machine this disk is added to

### 34.4.2 Get Disk Details (DRaaS)

To get the details of a specific disk in DRaaS Dashboard, use the following request:

```
GET /disks/:disk_id.json
```

**JSON Request Example**

```
```

Where:

- `API_KEY` - your API key

**JSON Output Example**

```
```

Where:

- `created_at` - the date when the disk was created
- `id` - the ID of the disk
number - the ordinal number of the disk on the list
primary - true, if this is a primary disk; otherwise, false
progress - disk replication readiness (in %)
replication_status - inactive, replication_in_progress, or in sync
size - the disk size (GB)
swap - true, if this is a swap disk; otherwise, false
updated_at - the date when the disk was updated for the last time
virtual_machine_id - the disk of the virtual machine this disk is added to

34.5 Events (DRaaS)

Events represent all recent transactions on DRaaS Dashboard alongside their statuses. You may get the list of all events in Dashboard or the details of a specific event.

34.5.1 Get List of Events (DRaaS)

To get the list of all events in DRaaS Dashboard, use the following request:

GET /events.json

JSON Request Example

```
```

To get the list of events on a specific cloud in DRaaS Dashboard, use the following request:

GET /clouds/:cloud_id/events.json

JSON Request Example

```
```

To get the list of events on a specific virtual machine in DRaaS Dashboard, use the following request:

GET /virtual-machines/:virtual-machine_id/events.json

JSON Request Example

```
```

Where:

API_KEY - your API key
**JSON Output Example**

```json
[
  {
    "created_at": "2020-07-16T08:08:23.428Z",
    "id": "c17ecb53-092f-4c98-8879-932f369e21a8",
    "label": "ReplicationProgressUpdateJob",
    "status": "complete",
    "target_id": "985ff773-1bce-4800-a910-51aba27a6f44",
    "target_type": "VirtualMachine",
    "updated_at": "2020-07-16T08:08:25.175Z"
  },
  {
    "created_at": "2020-07-16T06:34:22.646Z",
    "id": "d276c812-2263-4d6e-af15-c0a4c691a99",
    "label": "MetadataMonitoringJob",
    "status": "complete",
    "target_id": "985ff773-1bce-4800-a910-51aba27a6f44",
    "target_type": "VirtualMachine",
    "updated_at": "2020-07-16T06:34:28.415Z"
  }
]
```

Where:

- `created_at` - the date when the event was created
- `id` - the ID of the event
- `label` - the label of the event
- `status` - the event’s status: notice, alert, pending, running, complete, failed, canceled, or retry
- `target_id` - the ID of the requested event
- `target_type` - IPAddress, NetworkInterface, Disk, VirtualMachine, or ComputeZone
- `updated_at` - the date when the event was updated

### 34.5.2 Get Event Details (DRaaS)

To get the details of a specific event in DRaaS Dashboard, use the following request:

GET /events/:event_id.json

**JSON Request Example**

```bash
```

Where:

- `API_KEY` - your API key

**JSON Output Example**

```json
{"created_at": "2020-05-20T20:59:59.514Z",
"id": "deec49fd-c7ae-4723-b9d4-93f12ab33629",
"label": "Remove Virtual Machine",
"status": "complete",
"target_id": "null",
"target_type": "VirtualMachine",
"updated_at": "2020-05-20T21:00:05.591Z"}
```

Where:

- `created_at` - the date when the event was created
- `id` - the ID of the event
- `label` - the label of the event
- `status` - the event’s status: notice, alert, pending, running, complete, failed, canceled, or retry
- `target_id` - the ID of the requested event
34.6 Get DRaaS Dashboard Version

To check the version of your DRaaS Dashboard, use the following request:

GET /version.json

**JSON Request Example**


**JSON Output Example**

```
{"major":2,"minor":5,"patch":0,"pre":null,"build":"cf33ald4e8e4e654798903c8fec9ca70dff90ebf","full":"2.5.0+cf33ald4e8e4e654798903c8fec9ca70dff90ebf"}
```

Where:

- **major** - the major version of DRaaS Dashboard
- **minor** - the minor version (subversion) of DRaaS Dashboard
- **patch** - the version of the current patch

For example, if your DRaaS version is 2.8.1, 2 is the major version, 8 is the minor version (subversion), and 1 is patch.

- **pre** - pre-version for development purposes; most frequently, the value will be empty
- **build** - the version of the current build
- **full** - all listed above parameters gather into one identifier; used mainly by developers

34.7 IP Ranges (DRaaS)

This section contains the API requests you can use to manage the IP ranges in your cloud: all CRUD operations are available for IP ranges.

34.7.1 Get List of Cloud IP Ranges

To get the list of cloud IP ranges in DRaaS Dashboard, use the following request:

GET /clouds/:cloud_id/ip-ranges?page=PAGE.json

**JSON Request Example**

Where:

API_KEY - your API key

JSON Output Example

```json
[
  {
    "address": "56.215.5.8","cloud_id": "176c1fb2b9f","created_at": "2020-06-03T14:05:18.070Z","id": "ab8f-d9148f08a239","updated_at": "2020-06-03T14:05:18.070Z"},
  {
    "address": "99.138.137.2/24","cloud_id": "4258-8866-4c76c16b2b9f","created_at": "2020-06-03T14:00:50.867Z","id": "48-bf6e-03db51bdf17f","updated_at": "2020-06-03T14:00:50.867Z"}
]
```

Where:

address - the IP address

cloud_id - the ID of the cloud

created_at - the date when the cloud IP range was created

id - the ID of the cloud IP range

updated_at - the date when the cloud IP range was updated for the last time

34.7.2 Get Cloud IP Range Details

To get the details of a specific cloud IP range, use the following request:

GET /ip-ranges/:ip-range_id.json

JSON Request Example

```bash
```

Where:

API_KEY - your API key

JSON Output Example

```json
{
  "address": "69.168.237.0/24","cloud_id": "1b7b08e4-c836-4258-8866-4c76c16b2b9f","created_at": "2020-06-03T14:00:50.867Z","id": "4b42e9c9-0360-45d8-bf6e-03db51bdf17f","updated_at": "2020-06-03T14:00:50.867Z"
}
```

Where:

address - the IP address

cloud_id - the ID of the cloud

created_at - the date when the cloud IP range was created

id - the ID of the cloud IP range

updated_at - the date when the cloud IP range was updated for the last time

34.7.3 Create Cloud IP Range (DRaaS)

To create a cloud IP range, use the following request:

POST /clouds/:cloud_id/ip-ranges.json
34.7.4 Update Cloud IP Range (DRaaS)

Please note that for the update to take place, all affected replications have to be restarted.

To update cloud IP range, use the following request:

```
PATCH /ip-ranges/:ip-range_id.json
```

**JSON Request Example**

```
curl -v http://draas.io/api/3/ip-ranges/4b42e6e-03db51bfdf17f -X PATCH -d
  '{"address":"69.168.226.0/24"}' -H 'Accept: application/json' -H
  'Authorization: Bearer API_KEY' -H 'Content-Type: application/json'
```

Where:

- `address` - an IP address
- `API_KEY` - your API key

34.7.5 Delete Cloud IP Range (DRaaS)

Please note that for the update to take place, all affected replications have to be restarted.
To delete a cloud IP range, use the following request:
DELETE /ip-ranges/:ip-range_id.json

**JSON Request Example**

```bash
```

Where:

*API_KEY* - your API key

### 34.8 Locations (DRaaS)

You may use an API request to get the list of all compute zone locations in DRaaS Dashboard or get the details of a specific location.

#### 34.8.1 Get Locations List

To get the list of locations in DRaaS Dashboard, use the following request:

GET /locations.json

**JSON Request Example**

```bash
```

Where:

*API_KEY* - your API key

**JSON Output Example**

```json
[{
  "city": "Guabang",
  "country": "China",
  "created_at": "2019-09-23T13:49:23.175Z",
  "id": "533ad7c1-53fb-4619-894d-eadae6b37086",
  "latitude": "",
  "longitude": "",
  "updated_at": "2019-09-23T13:49:23.175Z"
}, {
  "city": "Paris",
  "country": "France",
  "created_at": "2019-07-04T08:38:00.725Z",
  "id": "386840e0-6d5c-46d4-8bff-54dad4552910",
  "latitude": "48.856663",
  "longitude": "2.351556",
  "updated_at": "2019-07-04T08:38:00.725Z"
}]
```

Where:

*created_at* - the date when the location was created

*id* - the ID of the location

*latitude* - the latitude value

*longitude* - the longitude value

*updated_at* - the date when the location was updated for the last time
34.8.2 Get Location Details
To get the details of a specific location in DRaaS Dashboard, use the following request:
GET /locations/:location_id.json

JSON Request Example

```bash
```

Where:

- **API_KEY** - your API key

JSON Output Example

```json
{"city":"Toronto","country":"Canada","created_at":"2018-10-16T06:26:17.808Z","id":"a4726808-4a3d-4e1e-893d-da312146243f","latitude":"43.684345","longitude":"-79.431292","updated_at":"2018-10-16T06:26:17.808Z"}
```

Where:

- `created_at` - the date when the location was created
- `id` - the ID of the location
- `latitude` - the latitude value
- `longitude` - the longitude value
- `updated_at` - the date when the location was updated for the last time

34.9 Networks (DRaaS)

In the DRaaS Dashboard, there may be three possible network types:

- **Replication** - used for the replication in all possible cases
- **Internal** - may be used for replication only in case there are no replication type networks added to the compute zone
- **Unused** - must not be used for replication even if it is the only available network in the compute zone

You may view networks, IP addresses, networks interfaces, and network links in your DRaaS Dashboard. Also, you may update networks and create or delete network links.

34.9.1 Get List of Networks (DRaaS)

To get the list of all networks in DRaaS Dashboard, use the following request:
GET /compute-zones/:compute-zone_id/networks.json

JSON Request Example

```bash
```
Where:

**API_KEY** - your API key

**JSON Output Example**

```json
[{
  "compute_zone_id": "ca640406-7249-41d1-9a93-e151a68c589c",
  "created_at": "2020-06-05T11:51:09.824Z",
  "id": "e6194725-6061-469f-9adf-e9f1c6a4fle",
  "label": "public_69.168.237.0/24",
  "remote_id": "2",
  "type": "replication",
  "updated_at": "2020-06-05T11:51:09.824Z",
},
{
  "compute_zone_id": "ca640406-7249-41d1-9a93-e151a68c589c",
  "created_at": "2020-06-05T11:51:09.821Z",
  "id": "cc898490-3aba-475f-ae44-d1ddf5925037",
  "label": "private_174_192.168.2.0/24",
  "remote_id": "3",
  "type": "replication",
  "updated_at": "2020-06-05T11:51:09.821Z"
}]
```

Where:

- **compute_zone_id** - the ID of the compute zone the network is associated with
- **created_at** - the date when the network was created
- **id** - the ID of the network
- **label** - the label of the network
- **remote_id** - the ID of the network in Control Panel
- **type**:
  - **replication** - network is used to set up replication stream, each VM needs to have at least one network interface attached to replication network
  - **internal** - the IP addresses on network interfaces of Virtual Machines that are attached to private networks are mapped 1-to-1 from client to provider side (internal networks should be linked by another API call before usage)
  - **unused** - network is not used, Virtual Machines should not have any network interfaces attached to unused networks
- **updated_at** - the date when the network was updated for the last time

**34.9.2 Get Network Details (DRaaS)**

To get the details of a specific network, use the following request:

```
GET /networks/:network_id.json
```

**JSON Request Example**

```bash
```

Where:

**API_KEY** - your API key

**JSON Output Example**
```json
{"compute_zone_id":"ca640406-7249-41d1-9a93-e151a68c589c","created_at":"2020-06-05T11:51:09.821Z","id":"cc898490-3aba-475f-ae44-d1df5925037","label":"private_174_192.168.2.0/24","remote_id":"3","type":"replication","updated_at":"2020-06-05T11:51:09.821Z"}
```

Where:
- **compute_zone_id** - the ID of the compute zone
- **created_at** - the date when the network was created
- **id** - the ID of the network
- **label** - the label of the network
- **remote_id** - the ID of the network in Control Panel
- **type**:
  - **replication** - network is used to set up replication stream, each VM needs to have at least one network interface attached to replication network
  - **internal** - IP addresses on network interfaces of virtual machines that are attached to private networks are mapped one-to-one from client to provider side (internal networks should be linked by another API call before usage)
  - **unused** - network is not used, virtual machines should not have any network interfaces attached to unused networks
- **updated_at** - the date when the network was updated

**34.9.3 Update Network (DRaaS)**

This request will fail if the network has any network links.

This request is used to update the network type. Once a compute zone is registered, all the networks attached to it are discovered automatically and get replication type. If a particular network needs to be assigned unused or internal type then this call should be used. To update a network in DRaaS Dashboard, use the following request:

PATCH /networks/:network_id.json

**JSON Request Example**

```bash
```

Where:
- **type**:
  - **replication** - network is used to set up replication stream, each virtual machine needs to have at least one network interface attached to replication network
  - **internal** - IP addresses on network interfaces of virtual machines that are attached to private networks are mapped one-to-one from client to provider side (internal networks should be linked by another API call before usage), or
- unused - network is not used, virtual machines should not have any network interfaces attached to unused networks

API_KEY - your API key

### 34.9.4 Get List of IP Addresses (DRaaS)

To get the list of IP addresses of a specific virtual machine in DRaaS Dashboard, use the following request:

GET /virtual-machines/:virtual-machine_id/ip-addresses.json

**JSON Request Example**

```sh
```

To get the list of IP addresses from a specific network interface in DRaaS Dashboard, use the following request:

GET /network-interfaces/:network-interface_id/ip-addresses.json

**JSON Request Example**

```sh
```

Where:

API_KEY - your API key

**JSON Output Example**

```
[{
"created_at":"2020-07-15T12:41:31.732Z","id":"d5e9ef8d-90d0-4e0b-a3dc-0bb333cc0d769","network_interface_id":"42f24ca5-1100-4991-a32e-cb1da0e78722","updated_at":"
```

Where:

created_at - the date when the IP address was added

id - the ID of the IP address

network_interface_id - the ID of the network interface

updated_at - the date when the IP address was updated for the last time

### 34.9.5 Get IP Address Details (DRaaS)

To get the details of a specific IP address in DRaaS Dashboard, use the following request:

GET /ip-addresses/:ip-address_id.json

**JSON Request Example**

```
["created_at":"2020-07-15T12:41:31.732Z","id":"d5e9ef8d-90d0-4e0b-a3dc-0bb333cc0d769","network_interface_id":"42f24ca5-1100-4991-a32e-cb1da0e78722","updated_at":"
```

Where:

created_at - the date when the IP address was added

id - the ID of the IP address

network_interface_id - the ID of the network interface

updated_at - the date when the IP address was updated for the last time
OnApp Cloud 6.3 API Guide

**34.9.6 Get Network Interfaces List (DRaaS)**

To get the list of network interfaces for a specific virtual machine in DRaaS Dashboard, use the following request:

GET /virtual-machines/:virtual-machine_id/network-interfaces.json

**JSON Request Example**

```
```

**Where:**

- **API_KEY** - your API key
- **created_at** - the date when the network interface was created
- **id** - the ID of the network interface
- **label** - the label of the network interface
- **network_id** - the ID of the network associated with this network interface
- **primary** - true, if this is a primary network interface; otherwise, false

**JSON Output Example**

```
[{"created_at":"2020-07-15T12:41:31.572Z","id":"42f24ca5-1100-4991-a32e-cbl1a0e78722","label":"Network Interface #0","network_id":"f7c67d1c-71f7-4b0b-9e12-bad0a889cc72","primary":true,"rate_limit":0,"updated_at":"2020-07-15T12:41:31.572Z","virtual_machine_id":"985ff773-1bce-4800-a910-51aba27a6f44"}]
```

**Where:**

- **API_KEY** - your API key
- **created_at** - the date when the network interface was added
- **id** - the ID of the IP address
- **network_interface_id** - the ID of the network interface
- **updated_at** - the date when the IP address was updated for the last time

**JSON Output Example**

```
```

**Where:**

- **API_KEY** - your API key
- **created_at** - the date when the IP address was added
- **id** - the ID of the IP address
- **network_interface_id** - the ID of the network interface
- **updated_at** - the date when the IP address was updated for the last time
rate_limit - the rate limit set for this network interface
updated_at - the date when the network interface was updated for the last time
virtual_machine_id - the ID of the virtual machine the network interface is associated with

34.9.7 Get Network Interface Details (DRaaS)
To get the details of a specific network interface in DRaaS Dashboard, use the following request:
GET /network/interfaces/:network_interface_id.json

JSON Request Example
```
```

Where:
API_KEY - your API key

JSON Output Example
```
{
  "created_at": "2020-07-15T12:41:31.572Z",
  "id": "42f24ca5-1100-4991-a32e-cb1da0e78722",
  "label": "Network Interface #0",
  "network_id": "f7c67d1c-71f7-4b0b-9e12-bad0a889cc72",
  "primary": true,
  "rate_limit": 0,
  "updated_at": "2020-07-15T12:41:31.572Z",
  "virtual_machine_id": "985ff773-1bce-4800-a910-51aba27a6f44"
}
```

Where:
created_at - the date when the network interface was created
id - the ID of the network interface
label - the label of the network interface
network_id - the ID of the network associated with this network interface
primary - true, if this is a primary network interface; otherwise, false
rate_limit - the rate limit set for this network interface
updated_at - the date when the network interface was updated for the last time
virtual_machine_id - the ID of the virtual machine the network interface is associated with

34.9.8 Network Links (DRaaS)
You may connect networks of internal type assigned to different compute zones in DRaaS Dashboard. You may use API requests in this section to view the existing connections, create the new ones, or remove the unnecessary links.

34.9.8.1 Get List of Network Links (DRaaS)
To get the list of all network links in DRaaS Dashboard, use the following request:
GET /network-links.json

JSON Request Example
To get the list of network links of a specific network in DRaaS Dashboard, use the following request:

GET /networks/:network_id/network-links.json

JSON Request Example

-H 'Accept: application/json' -H 'Authorization: Bearer API_KEY'

Where:

- **API_KEY** - your API key

JSON Output Example

```
```

Where:

- **client_id** - the ID of the client network
- **created_at** - the date when the network link was created
- **id** - the ID of the network link
- **provider_id** - the ID of the provider network
- **updated_at** - the date when the network link was updated for the last time

34.9.8.2 Get Network Link Details

To get the details of a specific network link in DRaaS Dashboard, use the following request:

GET /network-links/:network-link_id.json

JSON Request Example

curl -v http://draas.io/api/3/network-links/03eb44bcbdde866a
-H 'Accept: application/json' -H 'Authorization: Bearer API_KEY'

Where:

- **API_KEY** - your API key

JSON Output Example

```
```
```
{"client_id":"f34cbe40-2e99-41f3-8bc5-317d36e77ad3","created_at":"2020-07-14T11:39:08.368Z","id":"03eb44bc-ee5a-4702-94d1-51acbdde866a","provider_id":"1f5f3085-13c3-419a-8437-59c9d0d18019","updated_at":"2020-07-14T11:39:08.368Z"}
```

**Where:**

- **client_id** - the ID of the client network
- **created_at** - the date when the network link was created
- **id** - the ID of the network link
- **provider_id** - the ID of the provider network
- **updated_at** - the date when the network link was updated for the last time

### 34.9.8.3 Create Network Link (DRaaS)

To create a network link in DRaaS Dashboard, select one client and one provider network, and use the following request:

**POST /network-links.json**

- **Only networks of **Internal** type can be linked.**
- **This request will fail if the client network already has any network links in which it is a client.**

**JSON Request Example**

```
'{"client_id":"f34cbe40-2e99-41f3-8bc5-317d36e77ad3","provider_id":"1f5f3085-13c3-419a-8437-59c9d0d18019"}' -H 'Accept: application/json' -H 'Authorization: Bearer API_KEY' -H 'Content-Type: application/json'
```

**Where:**

- **client_id** - the ID of the client network
- **provider_id** - the ID of the provider network
- **API_KEY** - your API key

**Link to details of the newly registered Network Link can be found in the Location response header.**

### 34.9.8.4 Delete Network Link

To remove a network link from DRaaS Dashboard, use the following request:

**DELETE /network-links/:network-link_id.json**
This request will fail if there are any virtual machines with network interfaces attached to the client network of the network link.

**JSON Request Example**

```
curl -v http://draas.io/api/3/network-links/03eb441-51acbde866a -X DELETE
-H 'Accept: application/json' -H 'Authorization: Bearer API_KEY'
```

Where:

*API_KEY* - your API key

### 34.10 References (DRaaS)

You may use an API request to get a reference list for a compute zone, virtual machine, disk, network interface, or IP address. You can also get the list of all references in your DRaaS Dashboard.

#### 34.10.1 Get List of References

You may get a reference list for a compute zone, virtual machine, disk, network interface, or IP address.

To get the list of compute zone references in DRaaS Dashboard, use the following request:

```
GET /compute-zones/:compute-zone_id/references.json
```

**JSON Request Example**

```
curl -v http://draas.io/api/3/compute-zones/a3386a9c-ee59-40c2-a949-4e6c7d2a32/references?page=PAGE
-H 'Accept: application/json' -H 'Authorization: Bearer API_KEY'
```

To get the list of virtual machine references in DRaaS Dashboard, use the following request:

```
GET /virtual-machines/:virtual-machine_id/references.json
```

**JSON Request Example**

```
curl -v http://draas.io/api/3/virtual-machines/985ff0-51aba27a6f44/references?page=PAGE
-H 'Accept: application/json' -H 'Authorization: Bearer API_KEY'
```

To get the list of disk references in DRaaS Dashboard, use the following request:

```
GET /disks/:disk_id/references.json
```

**JSON Request Example**

```
curl -v http://draas.io/api/3/disks/:disk_id/references.json
-H 'Accept: application/json' -H 'Authorization: Bearer API_KEY'
```
To get the list of network interfaces references in DRaaS Dashboard, use the following request:

GET /network/interfaces/:network_interface_id/references.json

JSON Request Example

```bash
```

To get the list of IP address references in DRaaS Dashboard, use the following request:

GET /ip/addresses/:ip_address_id/references.json

JSON Request Example

```bash
```

Where:

- **API_KEY** - your API key

JSON Output Example

```json
[
  {
    "compute_zone_id": "a3386a9c-ee59-40c2-a949-4e6c7d02ca32",
    "created_at": "2020-07-15T12:43:19.654Z",
    "data": "value": "69.168.226.66",
    "id": "2bdb9091-500a-40dd-9f4d-a3daa0e78722",
    "remote_id": "64",
    "target_id": "d5e9ef8d-90d0-4e0b-a3dc0bb333ccd769",
    "target_type": "IPAddress",
    "updated_at": "2020-07-15T12:43:19.654Z"
  },
  {
    "compute_zone_id": "a3386a9c-ee59-40c2-a949-4e6c7d02ca32",
    "created_at": "2020-07-15T12:43:19.654Z",
    "data": "value": "69.168.226.66",
    "id": "2bdb9091-500a-40dd-9f4d-a3daa0e78722",
    "remote_id": "64",
    "target_id": "d5e9ef8d-90d0-4e0b-a3dc0bb333ccd769",
    "target_type": "IPAddress",
    "updated_at": "2020-07-15T12:43:19.654Z"
  }
]
```

Where:

- **compute_zone_id** - the ID of the compute zone
- **created_at** - the date when the reference was created
- **data** - this parameter is applicable for IP addresses, compute zones, and virtual machines; in other cases, it has a null value.

  - for virtual machines: **master** - true, if this is a master (initial) virtual machine; false, if this is a slave virtual machine
  - for compute zones: **master** - true, if this is a master (initial) compute zone; false, if this is a slave compute zone
for IP addresses: value - an IP address (e.g. 9.158.216.66)

id - the ID of the reference

remote_id - the ID of the reference in cloud Control Panel

target_id - the ID of the requested element (IP address, network interface, disk, virtual machine, or compute zone)

target_type - IPAddress, NetworkInterface, Disk, VirtualMachine, or ComputeZone

updated_at - the date when the reference was updated for the last time

34.10.2 Get Reference Details

To get the details of a specific reference in DRaaS Dashboard, use the following request:

GET /references/:reference_id.json

JSON Request Example


Where:

API_KEY - your API key

JSON Output Example

{"compute_zone_id":"a3386a9c-ee59-40c2-a949-4e6c7dce2a32","created_at":"2020-07-15T12:43:19.654Z","data":{"value":"69.168.226.66"},"id":"2bdb9091-500a-40dd-9f42-a3daa5a271fc","remote_id":"64","target_id":"d5e9e8d-90d0-4e0b-a3dc-0bb33cc5c769","target_type":"IPAddress","updated_at":"2020-07-15T12:43:19.654Z"}

Where:

compute_zone_id - the ID of the compute zone

created_at - the date when the reference was created

data - this parameter is applicable for IP addresses, compute zones, and virtual machines; in other cases, it has a null value.

for virtual machines: master - true, if this is a master (initial) virtual machine; false, if this is a slave virtual machine

for compute zones: master - true, if this is a master (initial) compute zone; false, if this is a slave compute zone

for IP addresses: value - an IP address (e.g. 9.158.216.66)

id - the ID of the reference

remote_id - the ID of the reference in cloud Control Panel

target_id - the ID of the requested element (IP address, network interface, disk, virtual machine, or compute zone)

target_type - IPAddress, NetworkInterface, Disk, VirtualMachine, or ComputeZone

updated_at - the date when the reference was updated for the last time
34.11 Users (DRaaS)

There are three types of roles in DRaaS Dashboard:

- **Administrator**: reserved for OnApp support engineers
- **Cloud owner**: created by support engineers; login credentials are provided by a support engineer
- **Cloud user**: is created automatically during cloud registration and can manage only one's own virtual server via the subdomain.draas.io dashboard (where subdomain stands for the user's VS IP address or hostname)

All CRUD operations are available for users.

34.11.1 Get List of Users (DRaaS)

To get the list of all users in DRaaS Dashboard, use the following request:

GET /users.json

**JSON Request Example**

```
```

Where:

- **API_KEY**: your API key

To get the list of all users for current VM, use the following request:

GET /clouds/:cloud_id/users.json

**JSON Request Example**

```
```

Where:

- **API_KEY**: your API key

**JSON Output Example**

```
[{
  "cloud_id":null,
  "created_at":"2020-06-01T16:41:56.596Z",
  "email":"roman.strazhnyk+1@onapp.com",
  "id":"aedf2ee5-e76f-47a9-bbef-db044a712101",
  "login":null,
  "name":"RS test",
  "remote_id":null,
  "role":"cloud_owner",
  "updated_at":"2020-06-01T16:41:56.849Z"},
{
  "cloud_id":null,
  "created_at":"2019-03-21T11:58:35.138Z",
  "email":"zara408201@gmail.com",
  "id":"077a32d8-5be8-4905-b25c-874834e63d3d",
  "login":null,
  "name":"Arsen Customer",
  "remote_id":null,
  "role":"cloud_owner",
  "updated_at":"2019-03-21T11:58:35.138Z"
}]
```

Where:

- **cloud_id**: the ID of the cloud where the user is registered
created_at - the date when the user is registered
email - user’s email address
id - the ID of the newly created user
login - the user’s login in cloud Control Panel. This field is not used for users with the Cloud Owner role and it is required for unprivileged users.
name - the user’s name
remote_id - the ID of the user in Control Panel
role - the user’s role (Cloud Owner by default)
updated_at - the date when the user was updated

34.11.2 Get User Details (DRaaS)
To get the details of a specific user in DRaaS Dashboard, use the following request:
GET /users/:user_id.json

JSON Request Example
```
```

Where:
API_KEY - your API key

JSON Output Example
```
{"cloud_id":null,"created_at":"2020-06-01T16:41:56.596Z","email":"rom@onapp.com","id":"aebfef- db044a712101","login":null,"name":"RS test","remote_id":null,"role":"cloud_owner","updated_at":"2020-06-01T16:41:56.849Z"}
```

Where:
cloud_id - the ID of the cloud
created_at - the date when the user was created
email - the user’s email address
id - the ID of the user
login - the user’s login in cloud Control Panel. This field is not used for users with the Cloud Owner role and it is required for unprivileged users.
name - the user’s name
remote_id - the ID of the user in Control Panel
role - the user’s role
updated_at - the date when the user was updated

34.11.3 Register New User
To register a new user in DRaaS Dashboard, use the following request:
POST /users.json
JSON Request Example

```
curl -v http://draas.io/api/3/users -X POST -d
   '{"email":"testemail1@gmail.com","name":"testuser","role":"cloud_owner"}'
   'Content-Type: application/json'
```

Where:

- **API_KEY** - your API key

JSON Output Example

```
{
  "cloud_id":null,
  "created_at":"2020-06-05T01:37:21.102Z",
  "email":"testemail1@gmail.com",
  "id":"0554220c-9c68-4e75-81aa-d09ba2413dc9",
  "login":null,
  "name":"testuser",
  "remote_id":null,
  "role":"cloud_owner",
  "updated_at":"2020-06-05T01:37:21.109Z"
}
```

Where:

- **cloud_id** - the ID of the cloud where the user is registered
- **created_at** - the date when the user is registered
- **email** - user’s email address
- **id** - the ID of the newly created user
- **login** - the user’s login
- **name** - the user’s name
- **remote_id** - identifier of the user in Cloud Control Panel. This field is not used for users with **cloud_owner** role and required for users with **unprivileged** role.
- **role** - the user’s role; **unprivileged** or **cloud_owner** (unprivileged by default)
- **updated_at** - the date when the user was updated for the last time

### 34.11.4 Update User (DRaaS)

To edit a specific user in DRaaS Dashboard, use the following request:

```
PATCH /users/:user_id.json
```

**JSON Request Example**

```
curl -v http://draas.io/api/3/users/f311938e0adc -X PATCH -d
   '{"email":"test@gmail.com","name":"testuser"}'
   'Content-Type: application/json'
```

Where:

- **email** - user’s email address
- **name** - user’s name
34.11.5 Delete User (DRaaS)

Only users that have no clouds and/or virtual machines can be removed.

To remove a user from DRaaS Dashboard, use the following request:

DELETE /users/:user_id.json

**JSON Request Example**

```
```

Where:

*API_KEY* - your API key

34.12 Virtual Machines (DRaaS)

Having virtual machines properly added and configured in your DRaaS Dashboard allows performing disaster recovery whenever necessary. All CRUD operations are possible for the virtual machines class.

34.12.1 Get List of Virtual Machines (DRaaS)

You may get the list of all virtual machines in DRaaS Dashboard, all VMs on a specific cloud, all VM from a specific compute zone, or all VMs from a specific location. Refer to the sections below to find the necessary API request.

34.12.1.1 Get List of All Virtual Machines

To get the list of all virtual machines in DRaaS Dashboard, use the following request:

GET /virtual-machines.json

**JSON Request Example**

```
```

Where:

*API_KEY* - your API key

*PAGE* (optional, default = 1) - pagination offset

*PER_PAGE* (optional, default = 10) - pagination window size

**JSON Output Example**
34.12.1.2 Get List of Virtual Machines on a Specific Cloud
To get the list of virtual machines on a specific cloud in DRaaS Dashboard, use the following request:

GET /clouds/:cloud_id/virtual-machines.json

**JSON Request Example**

```bash
```

Where:

- **API_KEY** - your API key
- **PAGE** (optional, default = 1) - pagination offset.
- **PER_PAGE** (optional, default = 10) - pagination window size.

**JSON Output Example**

```json
[{
  "created_at": "2020-07-14T13:10:56.768Z",
  "details": {
    "cpu_shares": 1,
    "cpus": 1,
    "hostname": "DraaSvm",
    "memory": 1024,
    "id": "7766c322-ff19-48dc-8277-a0d0406ac37",
    "label": "DraaSvm",
    "status": "replication_in_sync",
    "updated_at": "2020-07-14T13:43:57.175Z",
    "user_id": "a60be9e6-cbc9-46ae-a663-e4c4991e0117"
  }
}]
```

34.12.1.3 Get List of Virtual Machines from a Specific Compute Zone
To get the list of virtual machines from a specific compute zone in DRaaS Dashboard, use the following request:

GET /compute-zones/:compute-zone_id/virtual-machines.json

**JSON Request Example**

```bash
```

Where:

- **API_KEY** - your API key
- **PAGE** (optional, default = 1) - pagination offset.
- **PER_PAGE** (optional, default = 10) - pagination window size.

**JSON Output Example**

```json
[{
  "created_at": "2020-07-14T13:10:56.768Z",
  "details": {
    "cpu_shares": 1,
    "cpus": 1,
    "hostname": "DraaSvm",
    "memory": 1024,
    "id": "7766c322-ff19-48dc-8277-a0d0406ac37",
    "label": "DraaSvm",
    "status": "replication_in_sync",
    "updated_at": "2020-07-14T13:47:57.231Z",
    "user_id": "a60be9e6-cbc9-46ae-a663-e4c4991e0117"
  }
}]
```
34.12.1.4 Get List of Virtual Machines from a Specific Location
To get the list of virtual machines from a specific location, use the following request:

GET `/locations/:location_id/virtual-machines.json`

**JSON Request Example**

```bash
```

Where:

- **API_KEY** - your API key
- **PAGE** (optional, default = 1) - pagination offset.
- **PER_PAGE** (optional, default = 10) - pagination window size.

**JSON Output Example**

```json
[{
    "created_at": "2020-07-14T13:10:56.768Z",
    "details": {
        "cpu_shares": 1,
        "cpus": 1,
        "hostname": "DraaSvm",
        "memory": 1024,
        "id": "7766c322-ff19-48dc-8277-a06d0406ac37",
        "label": "DraaSvm",
        "status": "replication_in_sync",
        "updated_at": "2020-07-14T13:53:57.317Z",
        "user_id": "a60be9e6-46ae-46ae-a663-e4c49910117"
    }
}]
```

Where:

- **created_at** - the date when the virtual machine was created
- **details**:
  - **cpu_shares** - the number of CPU shares of this virtual machine
  - **hostname** - hostname of the virtual machine
  - **memory** - the amount of RAM of the virtual machine
  - **id** - the ID of the virtual machine
  - **label** - the label of the virtual machine
  - **updated_at** - the date when the virtual machine was updated for the last time
  - **user_id** - the ID of the virtual machine's owner
- **status** - one of the following:
  - **registration** - the virtual machine is being registered in the Dashboard
  - **registration_failure** - virtual machine failed to register (could be an incorrect configuration or unavailable provider compute zone)
  - **metadata_sync** - virtual machine metadata (disks, network interface, IP addresses) is being synchronized from the client to the provider compute zone
replication_healing - virtual machine’s data replication is being set up from the client to the provider compute zone
replication_in_progress - the virtual machine’s data is being replicated from the client to the provider compute zone
replication_in_sync - the virtual machine’s data is synchronized between the client and provider compute zones. The virtual machine is ready for clean failover.
failover_in_progress - the virtual machine is being failed over to the provider compute zone
failover_complete - the virtual machine is running in failover mode in the provider compute zone
failback_preparation - the virtual machine is being prepared to failback to the client compute zone
failback_metadata_sync - the virtual machine metadata changes are being synchronized from the provider to the client compute zone
failback_replication_healing - the virtual machine data replication is being set up from the provider to the client compute zone
failback_replication_in_progress - the virtual machine data is being replicated from the provider to the client compute zone
failback_replication_in_sync - the virtual machine data is synchronized between the provider and client compute zones. The virtual machine is ready for failback finalization.
failback_finalization - the virtual machine is being transferred back to the client compute zone
failback_cancellation - failback is canceled, the virtual machine is being transferred back to failover mode on the provider compute zone
destruction - the virtual machine is being destroyed

34.12.2 Get Virtual Machine Details (DRaaS)
To get the details of a specific virtual machine in DRaaS Dashboard, use the following request:

GET /virtual-machines/:virtual-machine_id.json

JSON Request Example

```
```

Where:

API_KEY - your API key

PAGE (optional, default = 1) - pagination offset.

PER_PAGE (optional, default = 10) - pagination window size.

JSON Output Example

```
{
  "created_at": "2020-07-14T13:10:56.768Z",
  "details": {
    "cpu_shares": 1,
    "cpus": 1,
    "hostname": "DraaSvm",
    "memory": 1024
  },
  "id": "7766c38277ac37",
  "label": "DraaSvm",
  "status": "replication_healing",
  "updated_at": "2020-07-14T13:10:56.768Z",
  "user_id": "a60be9e6-cbc9-46ae-a663-e4c4991e0117"
}
```
Where:

- `created_at` - the date when the virtual machine was created
- `details`:
  - `cpu_shares` - the number of CPU shares of this virtual machine
  - `hostname` - hostname of the virtual machine
  - `memory` - the amount of RAM of the virtual machine
- `id` - the ID of the virtual machine
- `label` - the label of the virtual machine
- `updated_at` - the date when the virtual machine was updated for the last time
- `user_id` - the ID of the virtual machine’s owner
- `status` - one of the following:
  - `registration` - the virtual machine is being registered in the Dashboard
  - `registration_failure` - virtual machine failed to register (could be an incorrect configuration or unavailable provider compute zone)
  - `metadata_sync` - virtual machine metadata (disks, network interface, IP addresses) is being synchronized from the client to the provider compute zone
  - `replication_healing` - virtual machine’s data replication is being set up from the client to the provider compute zone
  - `replication_in_progress` - the virtual machine’s data is being replicated from the client to the provider compute zone
  - `replication_in_sync` - the virtual machine’s data is synchronized between the client and provider compute zones. The virtual machine is ready for clean failover.
  - `failover_in_progress` - the virtual machine is being failed over to the provider compute zone
  - `failover_complete` - the virtual machine is running in failover mode in the provider compute zone
  - `failback_preparation` - the virtual machine is being prepared to failback to the client compute zone
  - `failback_metadata_sync` - the virtual machine metadata changes are being synchronized from the provider to the client compute zone
  - `failback_replication_healing` - the virtual machine data replication is being set up from the provider to the client compute zone
  - `failback_replication_in_progress` - the virtual machine data is being replicated from the provider to the client compute zone
  - `failback_replication_in_sync` - the virtual machine data is synchronized between the provider and client compute zones. The virtual machine is ready for failback finalization.
  - `failback_finalization` - the virtual machine is being transferred back to the client compute zone
  - `failback_cancellation` - failback is canceled, the virtual machine is being transferred back to failover mode on the provider compute zone
  - `destruction` - the virtual machine is being destroyed

34.12.3 Register Virtual Machine (DRaaS)

To register a new virtual machine in DRaaS Dashboard, use the following request:

```
POST /virtual-machines.json
```
### JSON Request Example

```
```

**Where:**
- `cloud_id` - the ID of the cloud (This field is not used if the call is made using cloud API key)
- `remote_id` - the ID of the virtual machine in the cloud Control Panel
- `API_KEY` - your API key

This request schedules virtual machine registration:
*RegisterVirtualMachine* event is scheduled and its ID is returned in X-Event-Id response header.

#### 34.12.4 Perform Virtual Machine Failover (DRaaS)

To perform virtual machine failover, use the following request:

```
POST /virtual-machines/:virtual-machine_id/failover.json
```

This request is allowed only if the virtual machine’s status is `replication_in_sync`, or the client compute zone is not available.

**JSON Request Example**

```
```

**Where:**
- `API_KEY` - your API key

#### 34.12.5 Start Virtual Machine Failback (DRaaS)

To start virtual machine failback, use the following request:

```
POST /virtual-machines/:virtual-machine_id/start-failback.json
```

This request is allowed only when the virtual machine is in `failover_complete` status.

**JSON Request Example**
### 34.12.6 Finalize Virtual Machine Failback (DRaaS)

To finalize virtual machine failback, use the following request:

```
POST /virtual-machines/:virtual-machine_id/finalize-failback.json
```

This request is allowed only when the virtual machine is in `failback_replication_in_sync` status.

**JSON Request Example**

```
```

**Where:**

*API_KEY - your API key*

### 34.12.7 Cancel Virtual Machine Failback (DRaaS)

To cancel virtual machine failback, use the following request:

```
POST /virtual-machines/:virtual-machine_id/cancel-failback.json
```

This request is only allowed when the virtual machine is in `failback_replication_in_sync, failback_replication_in_progress, failback_replication_healing` or `failback_metadata_sync` statuses.

**JSON Request Example**

```
```

**Where:**

*API_KEY - your API key*

### 34.12.8 Remove Virtual Machine (DRaaS)

To remove a virtual machine from DRaaS Dashboard, use the following request:
DELETE /virtual-machines/:virtual-machine_id.json

**JSON Request Example**


Where:

API_KEY - your API key

To **forcefully** remove a virtual machine from DRaaS Dashboard, use the following request:

DELETE /virtual-machines/:virtual-machine_id.json

**JSON Request Example**


Where:

API_KEY - your API key
35 Embed Statistics Charts

The following request allows to get an OnApp statistics chart HTML code that can be then used for embedding to third party applications.

1. Include jquery (if you didn't include it before in your code):

   http://onapp.test/assets/jquery.js

2. Include highcharts:

   http://onapp.test/assets/highcharts/highcharts.js

3. Get the required chart with the following API call:

   curl -i -X GET -u user:userpass
   http://onapp.test/virtual_machines/:virtual_machine_id/cpu_usage.chart

HTML Example
4. Embed the HTML to your page.
36 Errors

OnApp produces the list of errors that occur on a Control Panel. Using API requests you can view the list of errors and the details of a particular error. If set in your system configuration, the list of errors is sent to OnApp in a form of an encrypted email. If required, you can disable the sending of the error list from your CP in your system configuration.

36.1 Get List of Errors

To view the list of Control Panel errors, use the following request:

GET /sysadmin_tools/infrastructure/errors.xml
GET /sysadmin_tools/infrastructure/errors.json

XML Request Example


JSON Request Example


XML Output Example
<infrastructure_errors type="array">
<infrastructure_error>

<backtrace>/onapp/interface/lib/daemon/activity/hypervisor_stats_runner.rb:15:in `run'
/onapp/interface/lib/daemon/supervisors/stats_supervisor.rb:46:in operation'
/onapp/interface/lib/core_ext/eventmachine.rb:8:in `call'
/onapp/interface/lib/core_ext/eventmachine.rb:8:in `block in spawn_threadpool'</backtrace>

<counter type="integer">215</counter>

<created_at type="datetime">2015-10-28T13:27:26+02:00</created_at>
<error_class>NoMethodError</error_class>

<fingerprint>00c2afa57790ab756d637b13f67f4de2c7a7b444c9195d9144cd32dec96b19</fingerprint>

</infrastructure_error>

<infrastructure_error>

<backtrace>/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/action_view/path_set.rb:58:in `find'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/action_view/lookup_context.rb:122:in `find'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/action_view/renderer/partial_renderer.rb:339:in `find_template'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/action_view/renderer/partial_renderer.rb:333:in `find_partial'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/action_view/renderer/partial_renderer.rb:222:in `render'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/action_view/renderer/erb/erb.rb:41:in `render'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/action_controller/metal/flash.rb:15:in `render'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/abstract_controller/rendering.rb:110:in `_render_template'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/abstract_controller/metal/streaming.rb:225:in `_render_template'
/onapp/interface/vendor/bundle/ruby/2.1.0/gems/actionpack-3.2.22/lib/abstract_controller/renders.rb:103:in `render_to_body'</backtrace>

<counter type="integer">33</counter>

<created_at type="datetime">2015-11-09T14:22:35+02:00</created_at>
<error_class>ActionView::MissingTemplate</error_class>

<fingerprint>a5c65d66c07e3d3396c9c4b5170b79d4f6c88f441ff563b3230d8963d744961</fingerprint>

</infrastructure_error>

</infrastructure_errors>
Where:

- **backtrace** - the backtrace of the error
- **counter** - how many times the error has occurred
- **created_at** - the date when this record was created in database in the `[YYYY][MM][DD][hh][mm][ss]Z` format
- **error_class** - the class of the error
- **fingerprint** - the unique identifier of the error
- **id** - ID of the error
- **message** - the message that will be sent with this error
- **reported** - whether the error has been reported or not
- **updated_at** - the date when this record was updated in database in the `[YYYY][MM][DD][hh][mm][ss]Z` format

### 36.2 Get Error Details

To view the details of an application error, use the following request:

- GET `/sysadmin_tools/infrastructure/errors/:id.xml`
- GET `/sysadmin_tools/infrastructure/errors/:id.json`

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

---
<infrastructure_error>

<backtrace>/onapp/interface/lib/daemon/activity/hypervisor_stats_runner.rb:15:in `run'
/onapp/interface/lib/daemon/supervisors/stats_supervisor.rb:46:in `operation'
/onapp/interface/lib/core_ext/eventmachine.rb:8:in `call'
/onapp/interface/lib/core_ext/eventmachine.rb:8:in 'block in spawn_threadpool'</backtrace>

<counter type="integer">215</counter>

<created_at type="datetime">2015-10-28T13:27:26+02:00</created_at>
<error_class>NoMethodError</error_class>

<fingerprint>00c2afa57790ab756d637b13ffe4fde2c7a7b444c9195d9144cd32dec96b19</fingerprint>

<id type="integer">5</id>
<message>undefined method `id' for nil:NilClass</message>
<br reported type="boolean">false</reported>

<updated_at type="datetime">2015-11-09T14:26:00+02:00</updated_at>
</infrastructure_error>

Where:

backtrace - the backtrace of the error
counter - how many times the error has occurred
created_at - the date when this record was created in database in the [YYYY][MM][DD][hh][mm][ss]Z format
error_class - the class of the error
fingerprint - the unique identifier of the error
id - ID of the error
message - the message that will be sent with this error
reported - whether the error has been reported or not
updated_at - the date when this record was updated in database in the [YYYY][MM][DD][hh][mm][ss]Z format
37 Federation

OnApp Federation is a central trading system that connects Suppliers, Traders and Users to create the largest federated compute resource cloud in the world and gives each OnApp Hosting Partner a global reach and almost unlimited capacity. This chapter provides requests for suppliers and traders.

37.1 Add Zone to Federation

As a supplier, you can submit a zone to Federation. To add a zone, use the following request:

POST /federation/hypervisor_zones/:id/add.xml
POST /federation/hypervisor_zones/:id/add.json

XML Request Example

curl -i -X POST -H 'Content-type: application/xml' http://onapp.test/federation/hypervisor_zones/21/add.xml -u user:userpass -d '  
<hypervisor_zone><label>test</label><network_zone_label>netw</network_zone_label><data_store_zone_label>ds</data_store_zone_label><network_zone_id>146</network_zone_id><data_store_zone_id>145</data_store_zone_id><template_group_id>11</template_group_id><description>favorite</description><hypervisor_zone_pricing_attributes><cpu_max>123</cpu_max><cpu_on>12</cpu_on><cpu_off>489</cpu_off><cpu_priority_max>903</cpu_priority_max><cpu_priority_on>34</cpu_priority_on><cpu_priority_off>12</cpu_PRIORITY_OFF><memory_max>256</memory_max><memory_on>128</memory_on><memory_off>12</memory_off><hypervisor_zone_pricing_attributes><data_store_zone_pricing_attributes><disk_size_max>99</disk_size_max><disk_size_on>22</disk_size_on><disk_size_off>22</disk_size_off><data_read>11</data_read><data_write>90</data_write><input_requests>20</input_requests><output_requests>5</output_requests><network_zone_pricing_attributes><ip_addresses_max>90</ip_addresses_max><ip_addresses_on>25</ip_addresses_on><ip_addresses_off>20</ip_addresses_off><port_speed_max>100</port_speed_max><port_speed_on>10</port_speed_on><port_speed_off>10</port_speed_off><data_sent>1</data_sent><data_recv>10</data_recv><network_zone_pricing_attributes><user_virtual_server_pricing_attributes><auto_scaling_max>11</auto_scaling_max><auto_scaling>10</auto_scaling><template_backup_store_max>100</template_backup_store_max><backup_disk_size_max>10</backup_disk_size_max><backup_disk_size_on>10</backup_disk_size_on><backup_disk_size_off>10</backup_disk_size_off><user_virtual_server_pricing_attributes></user_virtual_server_pricing_attributes></hypervisor_zone>

JSON Request Example

```json

```

495
curl -i -X POST -H 'Content-type: application/json'
http://onapp.test/federation/hypervisor_zones/21/add.json -u user:userpass
-d '{"hypervisor_zone":{"label":"test", "network_zone_label":"netw", "data_store_zone_label":"ds", "network_zone_id":"146", "data_store_zone_id":"145", "template_group_id":"11", "description":"favorite group", "hypervisor_zone_pricing_attributes":{"cpu_max":"24", "cpu_on":"12", "cpu_off":"6", "cpu_priority_max":"50", "cpu_priority_on":"30", "cpu_priority_off":"10", "memory_max":"256", "memory_on":"124", "memory_off":"124"}, "data_store_zone_pricing_attributes":{"disk_size_max":"100", "disk_size_on":"60", "disk_size_off":"10", "data_read":8, "data_write":12, "input_requests":12, "output_requests":23}, "network_zone_pricing_attributes":{"ip_addresses_max":90, "ip_addresses_on":10, "ip_addresses_off":5, "port_speed_max":100, "port_speed":6, "data_rxed":100, "data_sent":100}, "user_virtual_server_pricing_attributes":{"auto_scaling_max":50, "auto_scaling":10, "template_backup_store_max":50, "template_backup_store":10, "backup_disk_size_max":100, "backup_disk_size":10, "template_disk_size_max":100, "template_disk_size":10})}

Where:

id* - hypervisor zone ID
label* - add an optional description.
network_zone_label* - give a label of the network zone
data_store_zone_label* - give a label of the data store zone
network_zone_id* - set ID of the network zone
data_store_zone_id* - set ID of the data store zone

the array of compute zone details:

template_group_id - the ID of the template group. All the templates which are added to this group will be available to traders who sign up for this federated zone.
description - add required description

the array of compute zone pricing attributes
cpu_max - the maximum number of CPU cores the users can request when signed up for this federated zone
cpu_on - the price per CPU core per hour, for VSs powered on
cpu_off - the price per CPU core per hour, for VSs powered off
cpu_priority_max - the maximum CPU priority % the users can request when signed up for this federated zone
cpu_priority_on - the prices per CPU priority % per hour, for VSs powered on
cpu_priority_off - the prices per CPU priority % per hour, for VSs powered off
memory_max - the maximum RAM users can request when signed up for this federated zone.
memory_on - the price per Mb/hour for RAM when a VS is on

memory_off - the price per Mb/hour for RAM when a VS is off

the array of data store zone pricing attributes
disk_size_max - the maximum disk size the users can request when signed up for this federated zone
disk_size_on - the price per GB of disk space for VSs powered on
disk_size_off - the price per GB of disk space for VSs powered off
data_read - the price for data read per Gb
data_write - the price for data written per GB
input_requests - the price for input requests. Input requests are measured in millions and priced
per million requests. 

output_requests - the price for output requests. Output requests are measured in millions and priced per million requests.

the array of network zone pricing attributes 

- ip_addresses_max - the maximum number of IP addresses the users can request when signed up for this federated zone.
- ip_addresses_on - the price per IP address/per hour for VSs powered on
- ip_addresses_off - the price per IP address/per hour for VSs powered off
- port_speed_max - the maximum port speed the users can request when signed up for this federated zone.
- port_speed - the price per Mb per second of port speed
- data_rxed - the price per GB of data received
- data_sent - the price per GB of data sent

the array of user virtual server pricing attributes 

- auto_scaling_max - the number of VSs using Autoscaling that users can request when signed up for this federated zone
- auto_scaling - the price for the VSs using Autoscaling (per VS)
- template_backup_store_max - the total amount of disk space customers can use for backups and templates when signed up for this federated zone
- template_backup_store - set price for backups & templates
- backup_disk_size_max - the total amount of backup disk size users may use
- backup_disk_size - set price for the backup disk size
- template_disk_size_max - the total amount of backup server space users can request
- template_disk_size - set price for template disk size.

37.2 Enable Federated Zone

As a supplier, you can enable a federated zone. To do so, use the following request:

POST /federation/hypervisor_zones/:id/activate.xml
POST /federation/hypervisor_zones/:id/activate.json

XML Request Example

```bash
curl -i -X POST 
http://onapp.test/federation/hypervisor_zones/12/activate.xml 
-u user:userpass -d ''
```

JSON Request Example

```bash
curl -i -X POST 
http://onapp.test/federation/hypervisor_zones/12/activate.json 
-u user:userpass -d ''
```

Where: 

- id" - hypervisor zone ID

37.3 Disable Federated Zone

As a supplier, you can disable a zone to prevent users from adding any more VSs to the federated zone. To disable a zone, use the following request:
POST /federation/hypervisor_zones/:id/deactivate.xml
POST /federation/hypervisor_zones/:id/deactivate.json

XML Request Example

```
curl -i -X POST
  http://onapp.test/federation/hypervisor_zones/:id/deactivate.xml -u
  user:userpass -d ''
```

JSON Request Example

```
curl -i -X POST
  http://onapp.test/federation/hypervisor_zones/:id/deactivate.json -u
  user:userpass -d ''
```

Where:

- **id** - hypervizor zone ID

### 3.7.4 Remove Zone from Federation

As a supplier, you can remove a zone from Federation. To do so, use the following request:

DELETE /federation/hypervisor_zones/:id/remove.xml
DELETE /federation/hypervisor_zones/:id/remove.json

XML Request Example

```
curl -i -X DELETE
  http://onapp.test/federation/hypervisor_zones/:id/remove.xml -u
  user:userpass
```

JSON Request Example

```
curl -i -X DELETE
  http://onapp.test/federation/hypervisor_zones/:id/remove.json -u
  user:userpass
```

Where:

- **id** - hypervizor zone ID

### 3.7.5 Get List of Federated Resources

As a trader, you can get the list of all compute zones in Federation with their details. To do so, use the following request:

GET /federation/hypervisor_zones/unsubscribed.xml
GET /federation/hypervisor_zones/unsubscribed.json

XML Request Example:
curl -i -X GET
http://onapp.test/federation/hypervisor_zones/unsubscribed.xml -u
user:userpass

JSON Request Example:

curl -i -X GET
http://onapp.test/federation/hypervisor_zones/unsubscribed.json -u
user:userpass

XML Output Example
<hypervisor_zones type="array">
    <hypervisor_zone>
        <label/>
        <provider_name>Technologies, Inc.</provider_name>
        <seller_page_url nil="true"/>
        <description>Technologies offers high performance, redundant compute, and storage.</description>
        <federation_id>resource:compute:onapp-sbWRG0iBtxX4s4XK9xOWB6TjY</federation_id>
        <country>United States</country>
        <city>Clara</city>
        <uptime_percentage type="integer">99</uptime_percentage>
        <tier>pro</tier>
        <latitude type="float">39.354108</latitude>
        <longitude type="float">-126.955236</longitude>
        <cpu_score type="integer">2111</cpu_score>
        <cpu_index type="integer">91</cpu_index>
        <bandwidth_score type="integer">386</bandwidth_score>
        <bandwidth_index type="integer">76</bandwidth_index>
        <disk_score type="integer">5</disk_score>
        <disk_index type="integer">33</disk_index>
        <cloud_index type="integer">67</cloud_index>
        <tier_cpu_index type="integer">91</tier_cpu_index>
        <tier_disk_index type="integer">33</tier_disk_index>
        <tier_bandwidth_index type="integer">76</tier_bandwidth_index>
        <tier_cloud_index type="integer">67</tier_cloud_index>
        <certificates type="array"/>
        <hypervisor_zone_pricing>
            <cpu_on>0.0001</cpu_on>
            <cpu_off>0.0001</cpu_off>
            <cpu_priority_on>0.0</cpu_priority_on>
            <cpu_priority_off>0.0</cpu_priority_off>
            <memory_on>0.00001</memory_on>
            <memory_off>0.00001</memory_off>
            <cpu_max nil="true"/>
            <cpu_priority_max nil="true"/>
        </hypervisor_zone_pricing>
        <data_store_zone_pricing>
            <disk_size_on>0.0005</disk_size_on>
            <disk_size_off>0.0005</disk_size_off>
            <data_read>0.0</data_read>
            <data_write>0.0</data_write>
            <input_requests>0.0</input_requests>
            <output_requests>0.0</output_requests>
            <disk_size_max nil="true"/>
        </data_store_zone_pricing>
        <network_zone_pricing>
            <ip_addresses_on>0.001</ip_addresses_on>
            <ip_addresses_off>0.001</ip_addresses_off>
            <port_speed>0.0</port_speed>
            <data_rxed>0.0</data_rxed>
            <data_sent>0.02</data_sent>
            <ip_addresses_max nil="true"/>
            <port_speed_max nil="true"/>
        </network_zone_pricing>
        <user_virtual_server_pricing>
            <auto_scaling>0.0</auto_scaling>
            <template_backup_store>0.0</template_backup_store>
            <backup>0.0</backup>
            <template>0.0</template>
            <auto_scaling_max nil="true"/>
            <template_backup_store_max nil="true"/>
            <backup_max nil="true"/>
            <template_max nil="true"/>
        </user_virtual_server_pricing>
    </hypervisor_zone>
</hypervisor_zones>
<tier_options>
  <ha type="boolean">true</ha>
  <sla type="boolean">true</sla>
  <storage_performance type="boolean">true</storage_performance>
  <backups type="boolean">true</backups>
  <templates type="boolean">true</templates>
  <windows_license type="boolean">true</windows_license>
  <ddos_protection type="boolean">true</ddos_protection>
  <ipv6 type="boolean">true</ipv6>
  <dns type="boolean">true</dns>
  <motion type="boolean">false</motion>
  <replication type="boolean">true</replication>
</tier_options>
</hypervisor_zone>
</hypervisor_zones>

37.6 Get Federated Resource Details

As a trader, you can find a particular federated resource to see its details. To do so, use the following request:

GET /federation/hypervisor_zones/unsubscribed.xml?q=query
GET /federation/hypervisor_zones/unsubscribed.json?q=query

XML Request Example:

curl -i -X GET
user:userpass

JSON Request Example

curl -i -X GET
user:userpass

Where:
query - part of the name of compute zone's label, location country or city

XML Output Example
<?xml version="1.0" encoding="UTF-8"?>
<federation_hypervisor_zones type="array">
  <federation_hypervisor_zone>
    <data_store_zone_label>Label</data_store_zone_label>
    <data_store_zone_pricing>
      <data_read type="float">10.0</data_read>
      <data_write type="float">10.0</data_write>
      <disk_size_max nil="true"/>
      <disk_size_off type="float">10.0</disk_size_off>
      <disk_size_on type="float">20.0</disk_size_on>
      <input_requests type="float">20.0</input_requests>
      <output_requests type="float">40.0</output_requests>
    </data_store_zone_pricing>
    <description>a test data store</description>
    <compute resource_type nil="true"/>
    <compute resource_zone_pricing>
      <cpu_max nil="true"/>
      <cpu_off type="float">3.0</cpu_off>
      <cpu_on type="float">5.0</cpu_on>
      <cpu_priority_max nil="true"/>
      <cpu_priority_off type="float">10.0</cpu_priority_off>
      <cpu_priority_on type="float">20.0</cpu_priority_on>
      <memory_max nil="true"/>
      <memory_off type="float">10.0</memory_off>
      <memory_on type="float">20.0</memory_on>
    </compute resource_zone_pricing>
    <label>zonex</label>
    <network_zone_label>wertwert</network_zone_label>
    <network_zone_pricing>
      <data_rxed type="float">20.0</data_rxed>
      <data_sent type="float">30.0</data_sent>
      <ip_addresses_max nil="true"/>
      <ip_addresses_off type="float">10.0</ip_addresses_off>
      <ip_addresses_on type="float">30.0</ip_addresses_on>
      <port_speed type="float">30.0</port_speed>
    </network_zone_pricing>
    <template_group_id nil="true"/>
    <user_virtual_server_pricing>
      <auto_scaling type="float">10.0</auto_scaling>
      <auto_scaling_max nil="true"/>
      <backup_disk_size type="float">30.0</backup_disk_size>
      <backup_disk_size_max nil="true"/>
      <template_backup_store type="float">10.0</template_backup_store>
      <template_backup_store_max nil="true"/>
      <template_disk_size type="float">30.0</template_disk_size>
      <template_disk_size_max nil="true"/>
    </user_virtual_server_pricing>
    <federation_id>resource:compute resource:onapp-9Qc9BTsFmT2mp5rHZumHXHWg</federation_id>
  </federation_hypervisor_zone>
</federation_hypervisor_zones>

37.7 Subscribe to Federated Zone

As a trader, you can subscribe to compute zones available in Federation. To do so, use the following request:

POST /federation/hypervisor_zones/:id/subscribe.xml
POST /federation/hypervisor_zones/:id/subscribe.json

XML Request Example
OnApp Cloud 6.3 API Guide

**37.8 Unsubscribe from Federated Zone**

As a trader, you can unsubscribe from a federated zone so that your users could create no more VSs in a zone. To do so, use the following request:

DELETE /federation/hypervisor_zones/:id/unsubscribe.xml
DELETE /federation/hypervisor_zones/:id/unsubscribe.json

**XML Request Example**

curl -i -X DELETE
http://onapp.test/federation/hypervisor_zones/:id/unsubscribe.xml -u
user:userpass -d ''

**JSON Request Example**

curl -i -X DELETE
http://onapp.test/federation/hypervisor_zones/:id/unsubscribe.json -u
user:userpass -d ''

Where:

id* - federated compute zone ID which you can find using GET request (e.g. resource:compute resource:onapp-KpY4wXq2N-9FERmf-SkwI2l1kQ4)

**37.9 Suspend Zone**

As a trader, you can prevent your users from creating any more VSs on a federated zone by suspending the zone. To suspend a zone, use the following request:

PUT /federation/hypervisor_zones/:id/close.xml
PUT /federation/hypervisor_zones/:id/close.json

**XML Request Example**

curl -i -X PUT
http://onapp.test/federation/hypervisor_zones/:id/close.xml -u
user:userpass -d ''

**JSON Request Example**

curl -i -X PUT
http://onapp.test/federation/hypervisor_zones/:id/close.json -u
user:userpass -d ''

Where:

id* - hypervizor zone ID
As a trader, you can unsuspend a suspended federated zone. To do so, use the following request:

PUT /federation/hypervisor_zones/:id/open.xml
PUT /federation/hypervisor_zones/:id/open.json

**XML Request Example**

```
curl -i -X PUT http://onapp.test/federation/hypervisor_zones/:id/open.xml
-u user:userpass -d ''
```

**JSON Request Example**

```
curl -i -X PUT http://onapp.test/federation/hypervisor_zones/:id/open.json
-u user:userpass -d ''
```

**Where**

- **id** - hypervisor zone ID
38 Firewall Rules for VSs

Firewall rules are applied to the VSs of your cloud to prevent unauthorized or unwanted requests to their network interfaces. You can configure your firewall to Accept/Drop specific request types. All methods are available for this class.

38.1 Get List of Firewall Rules

To get the list of firewall rules assigned to a VS, use the following request:

GET /virtual_machines/:virtual_machine_id/firewall_rules.xml
GET /virtual_machines/:virtual_machine_id/firewall_rules.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<firewall_rules>
  <firewall_rule>
    <position>1</position>
    <address></address>
    <created_at>2011-04-20T12:52:10+03:00</created_at>
    <command>ACCEPT</command>
    <updated_at>2011-04-20T12:52:10+03:00</updated_at>
    <port>21</port>
    <protocol>TCP</protocol>
    <id>1</id>
    <network_interface_id>5</network_interface_id>
    <comment>This is a comment</comment>
  </firewall_rule>
</firewall_rules>
```

Where:

- `position` – the rule priority
- `address` – the IP address for which this rule is active. If none is specified, all IPs will be subject to this rule.
- `created_at` – the date when the record in DB was created
- `command` – the action which will be performed with the IP specified by the `address` parameter
- `updated_at` – the date when the record was updated in DB
- `port` – the port for which this rule is active. If the field is empty, the rule will apply to all ports
- `protocol` – the IP protocol (TCP or UDP) for which this rule is active
id – the ID of this record

network_interface_id – the ID of a network interface for which this rule is active

comment - the comment added to the firewall rule

Page History
v. 6.1
• added the comment parameter

38.2 Apply Firewall Rule

Ensure that the following permissions are enabled before setting firewall rules for your virtual server:

• Create own firewall rules
• Destroy own firewall rules
• Read own firewall rules
• Update own firewall rules

To apply firewall rules for a virtual server, use the following request:

POST /virtual_machines/:virtual_machine_id/update_firewall_rules.xml
POST /virtual_machines/:virtual_machine_id/update_firewall_rules.json

XML Request Example:


JSON Request Example:


38.3 Add Firewall Rule

Ensure that the following permissions are enabled before setting firewall rules for your virtual server:

• Create own firewall rules
• Destroy own firewall rules
• Read own firewall rules
To add a firewall rule, use the request listed below. After you add a rule, you have to apply it to initiate a transaction responsible for running firewall rules. See the Apply Firewall Rule section for details.

POST /virtual_machines/:virtual_machine_id/firewall_rules.xml
POST /virtual_machines/:virtual_machine_id/firewall_rules.json

**XML Request example**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<?xml version="1.0" encoding="UTF-8"?><firewall_rule><address></address><command>DROP</command><port></port><protocol>TCP</protocol><network_interface_id>105</network_interface_id><comment>This is a comment</comment></firewall_rule>' --url http://onapp.test/virtual_machines/21/firewall_rules.xml
```

**JSON Request example**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"firewall_rule":{"address":","command":"DROP","protocol":"TCP","network_interface_id":"105","comment":"This is a comment","port":""}}' --url http://onapp.test/virtual_machines/21/firewall_rules.json
```

**Send the following parameters:**

- **address** - Set the IP address for which this rule is active.
  - Leave the empty field to apply this rule to all IPs
  - Enter hyphen-separated IPs to apply the rule to an IP range (e.g. 192.168.1.1-192.168.1.10)
  - Enter the IPs with slash to apply the rule to CIDR (e.g. 192.168.1.1/24)

- **command** - sets the command to ACCEPT or DROP the abovementioned IPs
  - port - sets the port addresses
    - Leave the empty field to apply the rule to all ports
    - Enter colon-separated ports to apply the rule to a port range (e.g. 1024:1028)
    - Enter comma-separated ports to apply the rule to the list of ports (e.g. 80,443,21)

- **protocol** - protocol type (TCP, UDP, DCCP, SCTP or ICMP)
  - **network_interface_id** - interface of the network

- **comment** - the comment to the firewall rule

**Protocols:**

For IPv4, only the ICMP, IPv6-ICMP, TCP, UDP, DCCP, SCTP protocols are available by default. However, if required, you can enable other protocols for IPv4.

1. Go to the /onapp/interface/config/network_protocols.yml file.
2. The list contains all protocols available (IPv4). Set 'true' for the required protocols.
3. Restart httpd by running one of the following commands:
4. The protocols you have enabled are now available while adding new firewall rules.

The following protocols can be enabled in the /onapp/interface/config/network_protocols.yml file:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>RDP</td>
<td>TLSP</td>
<td>AX.25</td>
<td></td>
</tr>
<tr>
<td>HOPOPT</td>
<td>IRTP</td>
<td>SKIP</td>
<td>IPIP</td>
<td></td>
</tr>
<tr>
<td>ICMP</td>
<td>ISO-TP4</td>
<td>CFTP</td>
<td>MICP</td>
<td></td>
</tr>
<tr>
<td>IGMP</td>
<td>NETBLT</td>
<td>SAT-EXPAK</td>
<td>SCC-SP</td>
<td></td>
</tr>
<tr>
<td>GGP</td>
<td>MFE-NSP</td>
<td>KRYPTOLAN</td>
<td>ETHERIP</td>
<td></td>
</tr>
<tr>
<td>IP-ENCAP</td>
<td>MERIT-INP</td>
<td>RVD</td>
<td>ENCAP</td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>DCCP</td>
<td>IPPC</td>
<td>GMTP</td>
<td></td>
</tr>
<tr>
<td>TCP</td>
<td>3PC</td>
<td>SAT-MON</td>
<td>IFMP</td>
<td></td>
</tr>
<tr>
<td>CBT</td>
<td>IDPR</td>
<td>VISA</td>
<td>PNNI</td>
<td></td>
</tr>
<tr>
<td>EGP</td>
<td>XTP</td>
<td>IPCV</td>
<td>PIM</td>
<td></td>
</tr>
<tr>
<td>IGP</td>
<td>DDP</td>
<td>CPNX</td>
<td>ARIS</td>
<td></td>
</tr>
<tr>
<td>BBN-RCC-MON</td>
<td>IDPR-CMTP</td>
<td>CPHB</td>
<td>SCPS</td>
<td></td>
</tr>
<tr>
<td>NVP-II</td>
<td>TP</td>
<td>WSN</td>
<td>QNX</td>
<td></td>
</tr>
<tr>
<td>PUP</td>
<td>IL</td>
<td>PVP</td>
<td>A/N</td>
<td></td>
</tr>
<tr>
<td>ARGUS</td>
<td>SDRP</td>
<td>BR-SAT-MON</td>
<td>IPComp</td>
<td></td>
</tr>
<tr>
<td>EMCON</td>
<td>IDRP</td>
<td>SUN-ND</td>
<td>SNP</td>
<td></td>
</tr>
<tr>
<td>XNET</td>
<td>RSVP</td>
<td>WB-MON</td>
<td>Compaq-Peer</td>
<td></td>
</tr>
<tr>
<td>CHAOS</td>
<td>GRE</td>
<td>WB-EXPAK</td>
<td>IPX-in-IP</td>
<td></td>
</tr>
<tr>
<td>UDP</td>
<td>DSR</td>
<td>ISO-IP</td>
<td>VRRP</td>
<td></td>
</tr>
<tr>
<td>MUX</td>
<td>BNA</td>
<td>VMTP</td>
<td>PGM</td>
<td></td>
</tr>
<tr>
<td>DCN-MEAS</td>
<td>ESP</td>
<td>SECURE-VMTP</td>
<td>L2TP</td>
<td></td>
</tr>
<tr>
<td>HMP</td>
<td>AH</td>
<td>VINES</td>
<td>DDX</td>
<td></td>
</tr>
<tr>
<td>PRM</td>
<td>i-NLSP</td>
<td>TTP</td>
<td>IATP</td>
<td></td>
</tr>
<tr>
<td>XNS-IDP</td>
<td>SWIPE</td>
<td>NSFNET-IGP</td>
<td>STP</td>
<td></td>
</tr>
<tr>
<td>TRUNK-1</td>
<td>NARP</td>
<td>DGP, TCF</td>
<td>SRP</td>
<td></td>
</tr>
<tr>
<td>TRUNK-2</td>
<td>MOBILE</td>
<td>EIGRP</td>
<td>UTI</td>
<td></td>
</tr>
<tr>
<td>LEAF-1</td>
<td>HIP</td>
<td>OSPFIGP</td>
<td>SMP</td>
<td></td>
</tr>
<tr>
<td>LEAF-2</td>
<td>manet</td>
<td>Sprite-RPC</td>
<td>SM</td>
<td></td>
</tr>
<tr>
<td>RSVP-E2E-IGNORE</td>
<td>MPLS-in-IP</td>
<td>LARP</td>
<td>PTP</td>
<td></td>
</tr>
<tr>
<td>FC</td>
<td>UDPLite</td>
<td>MTP</td>
<td>ISIS</td>
<td></td>
</tr>
</tbody>
</table>
38.4 Edit Firewall Rule

Ensure that the following permissions are enabled before setting firewall rules for your virtual server:

- Create own firewall rules
- Destroy own firewall rules
- Read own firewall rules
- Update own firewall rules

To edit a firewall rule, use the following request:

PUT /virtual_machines/:virtual_machine_id/firewall_rules/:id.xml
PUT /virtual_machines/:virtual_machine_id/firewall_rules/:id.json

**XML Request Example**

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
'"firewall_rule":{"address":"192.168.128.133","command":"ACCEPT","port":70,"protocol":"TCP","network_interface_id":105,"comment":"This is a comment"}"
http://onapp.test/virtual_machines/23/firewall_rules/211.xml

**JSON Request Example**

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
'{"firewall_rule":{"address":"192.168.128.133","command":"ACCEPT","port":70,"protocol":"TCP","network_interface_id":105,"comment":"This is a comment"}}'
-url
http://onapp.test/virtual_machines/23/firewall_rules/211.json

**Where:**

- `address` - Set the IP address for which this rule is active.
- Leave the empty field to apply this rule to all IPs
- Enter hyphen-separated IPs to apply the rule to an IP range (e.g. 192.168.1.1-192.168.1.10)
- Enter the IPs with slash to apply the rule to CIDR (e.g. 192.168.1.1/24)
**command** - sets the command to ACCEPT or DROP the abovementioned IPs

**port** - sets the port addresses
- Leave the empty field to apply the rule to all ports
- Enter colon-separated ports to apply the rule to a port range (e.g. 1024:1028)
- Enter comma-separated ports to apply the rule to the list of ports (e.g. 80,443,21)

**protocol** - protocol type (TCP or UDP)

**network_interface_id** - interface of the network

💡 **comment** - the comment added to the firewall rule

You will get a 204 status response on success, and 404 if there is no firewall rule with a requested ID or you entered incorrect URL.

### Protocols:

For IPv4, only the ICMP, IPV6-ICMP, TCP, UDP, DCCP, SCTP protocols are available by default. However, if required, you can enable other protocols for IPv4.

1. Go to the /onapp/interface/config/network_protocols.yml file.
2. The list contains all protocols available (IPv4). Set 'true' for the required protocols.
3. Restart httpd by running one of the following commands:

   ```
   service httpd restart
   ```

   or

   ```
   /etc/init.d/httpd restart
   ```

4. The protocols you have enabled are now available while adding new firewall rules.

The following protocols can be enabled in the /onapp/interface/config/network_protocols.yml file:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>RDP</td>
<td>TLSP</td>
<td>AX.25</td>
<td></td>
</tr>
<tr>
<td>HOPOPT</td>
<td>IRTP</td>
<td>SKIP</td>
<td>IPIP</td>
<td></td>
</tr>
<tr>
<td>ICMP</td>
<td>ISO-TP4</td>
<td>CFTP</td>
<td>MICP</td>
<td></td>
</tr>
<tr>
<td>IGMP</td>
<td>NETBLT</td>
<td>SAT-EXPAK</td>
<td>SCC-SP</td>
<td></td>
</tr>
<tr>
<td>GGP</td>
<td>MFE-NSP</td>
<td>KRYPTOLAN</td>
<td>ETHERIP</td>
<td></td>
</tr>
<tr>
<td>IP-ENCAP</td>
<td>MERIT-ISP</td>
<td>RVD</td>
<td>ENCAP</td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>DCCP</td>
<td>IPPC</td>
<td>GMTP</td>
<td></td>
</tr>
<tr>
<td>TCP</td>
<td>3PC</td>
<td>SAT-MON</td>
<td>IFMP</td>
<td></td>
</tr>
<tr>
<td>CBT</td>
<td>IDPR</td>
<td>VISA</td>
<td>PNNI</td>
<td></td>
</tr>
<tr>
<td>EGP</td>
<td>XTP</td>
<td>IP1V</td>
<td>PIM</td>
<td></td>
</tr>
<tr>
<td>IGP</td>
<td>DDP</td>
<td>IPCV</td>
<td>ARIS</td>
<td></td>
</tr>
<tr>
<td>BBN-RCC-MON</td>
<td>IDPR-CMTP</td>
<td>CPNX</td>
<td>SCPS</td>
<td></td>
</tr>
<tr>
<td>NVP-II</td>
<td>TP</td>
<td>CPHB</td>
<td>QNX</td>
<td></td>
</tr>
</tbody>
</table>
Ensure that the following permissions are enabled before setting firewall rules for your virtual server:

- **Create own firewall rules**
- **Destroy own firewall rules**
- **Read own firewall rules**
- **Update own firewall rules**

To delete a firewall rule, use the following request:

```
DELETE /virtual_machines/:virtual_machine_id/firewall_rules/:id.xml
DELETE /virtual_machines/:virtual_machine_id/firewall_rules/:id.json
```

**XML Request Example**
To destroy several firewall rules, use the following request:

```
DELETE /virtual_machines/:virtual_machine_id/firewall_rules/destroy_batch?ids=1,2,3.xml
DELETE /virtual_machines/:virtual_machine_id/firewall_rules/destroy_batch?ids=1,2,3.json
```

**XML Request Example**

```
curl -X DELETE -u user:userpass
```

**JSON Request Example**

```
curl -X DELETE -u user:userpass
```

You will get a 204 status response on success, and 404 if there is no firewall rule with a requested ID or you entered incorrect URL.

### 38.6 Change Firewall Rule Position

Ensure that the following permissions are enabled before setting firewall rules for your virtual server:

- *Create own firewall rules*
- *Destroy own firewall rules*
- *Read own firewall rules*
- *Update own firewall rules*
To change the position of the firewall rule, use the following request:

**GET**
/virtual_machines/:virtual_machine_id/firewall_rules/:firewall_rule_id/move.xml
**GET**
/virtual_machines/:virtual_machine_id/firewall_rules/:firewall_rule_id/move.json

**XML Request Example**

```bash
curl -X GET -u user:userpass
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example:**

```bash
curl -X GET -u user:userpass
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **virtual_machine_id** - ID of a virtual server the firewall rule belongs to
- **firewall_rule_id** - ID of the firewall rule
- **position** - specify the position change: up or down

## 38.7 Set Default Firewall Rules

Ensure that the following permissions are enabled before setting firewall rules for your virtual server:

- **Create own firewall rules**
- **Destroy own firewall rules**
- **Read own firewall rules**
- **Update own firewall rules**

To set default firewall rules for a VS (either DROP or ACCEPT), you need to set the rule for each network interface the VS is using. To do so, check the network interface ID and use the following request:

**PUT**
/virtual_machines/:virtual_machine_id/network_interfaces/:id.xml
**PUT**
/virtual_machines/:virtual_machine_id/network_interfaces/:id.json

**XML Request Example**

JSON Request Example


Where:

default_firewall_rule* - set default firewall rule for the particular VS network interface – either DROP or ACCEPT
39 Firewalls

Firewalls are used for managing VLANs and route VS networking traffic in and out of OnApp.

39.1 Get List of Firewalls

To view the list of firewalls, use the following request:

GET /settings/firewalls.xml
GET /settings/firewalls.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```xml
<firewalls type="array">
<firewall>
<created_at type="datetime">2012-08-30T17:51:28+03:00</created_at>
<firewall_cluster_id type="integer" nil="true"/>
<id type="integer">1</id>
<inside_cidr type="integer">24</inside_cidr>
<inside_interface>eth1</inside_interface>
<inside_ip_address>192.168.1.1</inside_ip_address>
<name_of_default_rule/>
<outside_cidr type="integer">24</outside_cidr>
<outside_gateway_address>87.116.6.254</outside_gateway_address>
<outside_interface>eth0</outside_interface>
<outside_ip_address>87.116.6.249</outside_ip_address>
<password>pass</password>
<updated_at type="datetime">2012-08-30T18:16:57+03:00</updated_at>
<username>user</username>
</firewall>
</firewalls>
```

Where:

- **created_at** — the date when the rule was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** — the date when the rule was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **firewall_cluster_id** - ID of a firewall cluster
- **id** - ID of the Firewall rule
- **inside_cidr** - internal IP address prefix size
- **inside_interface** - interface used for managing firewall via CP
**inside_ip_address** - IP address used to manage firewall via CP

**name_of_default_rule** - default firewall settings for new VSs (DROP/ACCEPT)

**outside_cidr_type** - external IP address prefix size

**outside_gateway_address** - external gateway address

**outside_interface** - external firewall interface

**outside_ip_address** - external firewall IP

**preshared_cluster_secret** - shared key for heartbeat authentication

**password** - password for remote firewall management

**username** - specify username for the remote firewall management

### 39.2 Get Firewall Details

To get details of a particular firewall, use the following request:

GET /settings/firewalls/:id.xml
GET /settings/firewalls/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```xml
<firewall>
  <created_at type="datetime">2012-08-30T17:51:28+03:00</created_at>
  <firewall_cluster_id type="integer" nil="true"/>
  <id type="integer">1</id>
  <inside_cidr type="integer">24</inside_cidr>
  <inside_interface>eth1</inside_interface>
  <inside_ip_address>192.168.1.1</inside_ip_address>
  <name_of_default_rule/>
  <outside_cidr_type="integer">24</outside_cidr_type>
  <outside_gateway_address>87.116.6.254</outside_gateway_address>
  <outside_interface>eth0</outside_interface>
  <outside_ip_address>87.116.6.249</outside_ip_address>
  <password>pass</password>
  <preshared_cluster_secret nil="true"/>
  <updated_at type="datetime">2012-08-30T18:16:57+03:00</updated_at>
  <username>user</username>
</firewall>
```

Where:

**created_at** — the date when the rule was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at — the date when the rule was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

firewall_cluster_id - ID of the firewall cluster

id - ID of the Firewall rule

inside_cidr - internal CIDR notation

inside_interface - interface used for managing firewall via CP

inside_ip_address - IP address used to manage firewall via CP

name_of_default_rule - default firewall settings for new VSs (DROP/ACCEPT)

outside_cidr_type - external CIDR notation

outside_gateway_address - external gateway address

outside_interface - external firewall interface

outside_ip_address - external firewall IP

preshared_cluster_secret - shared key for heartbeat authentication

password - password for remote firewall management

username - specify username for the remote firewall management

39.3 Add Firewall

To add a new firewall, use the following request:

POST /settings/firewalls.xml
POST /settings/firewalls.json

XML Request Example

curl -i POST -d '  <firewall><inside_ip_address>192.168.1.34</inside_ip_address><inside_cidr>22</inside_cidr><inside_interface>eth0</inside_interface><outside_ip_address>192.168.2.35</outside_ip_address><outside_cidr>24</outside_cidr><outside_interface>eth1</outside_interface><outside_gateway_address>192.168.0.0</outside_gateway_address><name_of_default_rule></name_of_default_rule><username>admin</username><password>tryrgfdghetrj</password></firewall>' -u user:userpass http://onapp.test/settings/firewalls.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml'

JSON Request Example

curl -i POST -d '{  "firewall":{
"inside_ip_address":"192.168.1.34",
"inside_cidr":"22",
"inside_interface":"eth0",
"outside_ip_address":"192.168.2.35",
"outside_cidr":"24",
"outside_interface":"eth1",
"outside_gateway_address":"192.168.0.0",
"name_of_default_rule":"",
"username":"admin",

Where:

inside_cidr - internal IP address prefix size

inside_interface - interface used for managing firewall via CP
inside_ip_address - IP address used to manage firewall via CP
name_of_default_rule - default firewall settings for new virtual servers (DROP/ACCEPT)
outside_cidr_type - external IP address prefix size
outside_gateway_address - external gateway address
outside_interface - external firewall interface
outside_ip_address - external firewall IP
password - password for remote firewall management
username - specify username for the remote firewall management

39.4 Edit Firewall

To edit a firewall, use the following request:

PUT /settings/firewalls/:id.xml
PUT /settings/firewalls/:id.json

XML Request Example

```
curl -i -X PUT -d
'<?xml version="1.0"?>
<firewall>
<inside_ip_address>192.168.1.34</inside_ip_address>
<outside_ip_address>192.168.2.35</outside_ip_address>
<outside_gateway_address>192.168.0.0</outside_gateway_address>
<name_of_default_rule/>
<username>admin</username>
<password>tryrgfdghetrj</password>
```

JSON Request Example

```
curl -i -X PUT -d '{
"firewall":{
"inside_ip_address":"192.168.1.34",
"outside_ip_address":"192.168.2.35",
"outside_gateway_address":"192.168.0.0",
"name_of_default_rule":null,
"username":"admin",
"password":"tryrgfdghetrj"
}
```

Where:
inside_cidr - internal IP address prefix size
inside_interface - interface used for managing firewall via CP
inside_ip_address - IP address used to manage firewall via CP
name_of_default_rule - default firewall settings for new virtual servers (DROP/ACCEPT)
outside_cidr_type - external IP address prefix size
outside_gateway_address - external gateway address
outside_interface - external firewall interface
outside_ip_address - external firewall IP
password - password for remote firewall management
**username** - specify username for the remote firewall management

### 39.5 Delete Firewall

To delete a firewall, use the following request:

```
DELETE /settings/firewalls/:id.xml
DELETE /settings/firewalls/:id.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

Returns HTTP 204 response on successful deletion, or HTTP 404 when a firewall with the ID specified is not found, or the URL requested is incorrect.
40 Hardware Info

OnApp provides an overview of hardware that is used by compute resources and backup servers available in your cloud. In this chapter, you can find information on how to view hardware info details and manage custom fields via API.

40.1 Get Hardware Info Details

To get hardware info details for a compute resource or backup server, use the following request:

GET /settings/:target/:target_id/hardware_info.xml
GET /settings/:target/:target_id/hardware_info.json

Where target could one of the following values, depending on the resource type:

- **hypervisors** - to get hardware info details for compute resources
- **backup_servers** - to get hardware info details for backup servers

**XML Request Example**

```bash
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:pass
http://onapp.test/settings/hypervisors/1/hardware_info.xml
```

**JSON Request Example**

```bash
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:pass
http://onapp.test/settings/hypervisors/1/hardware_info.json
```

**XML Output Example**
```xml
<hardware_info>
  <id>1</id>
  <cpu_sockets type="array">
    <hardware_info_cpu_socket_serializer>Intel(R) Core(TM) 2 Duo CPU</hardware_info_cpu_socket_serializer>
  </cpu_sockets>
  <memory_slots type="array">
    <hardware_info_memory_slot_serializer>Unknown, 800 MHz, 2048 MB</hardware_info_memory_slot_serializer>
    <hardware_info_memory_slot_serializer>Unknown, 800 MHz, 2048 MB</hardware_info_memory_slot_serializer>
  </memory_slots>
  <nics type="array">
    <hardware_info_nic_serializer>Intel Corporation 82574L Gigabit Network Connection</hardware_info_nic_serializer>
  </nics>
  <disks type="array">
    <hardware_info_disk_serializer>ST3500418AS 466 GB</hardware_info_disk_serializer>
  </disks>
  <bios>
    Award Software International, Inc., F4, release date: 07/22/2009
  </bios>
  <manufacturer>G41M-ES2L Gigabyte Technology Co., Ltd.</manufacturer>
  <created_at type="dateTime">2017-10-06T15:20:35+03:00</created_at>
  <updated_at type="dateTime">2018-02-01T17:40:43+02:00</updated_at>
  <uptime_custom_fields type="array">
    <uptime_custom_field>
      <custom_field_1>custom_value</custom_field_1>
    </uptime_custom_field>
  </uptime_custom_fields>
  <target_id type="integer">1</target_id>
  <target_type>Hypervisor</target_type>
</hardware_info>
```

**Where:**

- **id** - the ID of the hardware info
- **cpu_sockets** - the array of CPU sockets available for the compute resource/backup server
- **memory_slots** - the array of memory slots available for the compute resource/backup server
- **nics** - the array of network cards available for the compute resource/backup server
- **disks** - the array of hard disk drives available for the compute resource/backup server
- **bios** - the system BIOS, its serial number and release date
- **manufacturer** - the manufacturer and model of the motherboard
- **created_at** - the time when the hardware info was created
- **updated_at** - the time when the hardware info was updated
- **uptime_custom_fields** - the array of custom fields added to the hardware info
- **target_id** - the ID of the target compute resource/backup server
- **target_type** - the type of the resource (compute resource or backup server)
40.2 Update Hardware Info

To update hardware info for a compute resource or backup server, use the following request:

```
PUT /settings/:target/:target_id/hardware_info.xml
PUT /settings/:target/:target_id/hardware_info.json
```

Where `target` could one of the following values, depending on the resource type:

- **hypervisors** - to get hardware info details for compute resources
- **backup_servers** - to get hardware info details for backup servers

**XML Request Example**

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass
http://onapp.test/settings/hypervisors/1/hardware_info.xml
```

**JSON Request Example**

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass
http://onapp.test/settings/hypervisors/1/hardware_info.json
```

Returns **HTTP 200 OK** status when the request is successfully completed.

40.3 Get Hardware Info Custom Fields

To get hardware info custom fields for a compute resource or backup server, use the following request:

```
GET /settings/:target/:target_id/hardware_info/custom_fields.xml
GET /settings/:target/:target_id/hardware_info/custom_fields.json
```

Where `target` could one of the following values, depending on the resource type:

- **hypervisors** - to get hardware info custom fields for compute resources
- **backup_servers** - to get hardware info custom fields for backup servers

**XML Request Example**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass
http://onapp.test/settings/hypervisors/1/hardware_info/custom_fields.xml
```

JSON Request Example

```
```

XML Output Example

```
<objects type="array">
  <object>
    <label>CPU</label>
    <children type="array">
      <child>
        <label>Slot 0: Intel(R) Core(TM)2 Duo CPU</label>
        <children type="array">
          <child>
            <slot_id type="integer">0</slot_id>
            <parent_id>cpusockets</parent_id>
            <relations type="array">
              <relation>
                <label>1</label>
                <value>123</value>
                <id type="integer">0</id>
              </relation>
            </relations>
          </child>
        </children>
      </child>
    </children>
    <relations type="array"/>
  </object>
  <object>...
</objects>
```

Where:

- **objects** - the array of objects (components) on which the hardware info is collected. The hardware info is collected for the following objects:
  - **Summary** - includes basic information about a compute resource/backup server
  - **CPU** - lists CPU slots available for a compute resource/backup server
  - **RAM** - lists RAM slots available for a compute resource/backup server
  - **HD** - lists hard disk slots available for a compute resource/backup server
  - **Networks** - lists network slots available for a compute resource/backup server

- **label** - the name of the object (Summary, CPU, RAM, HD, or Networks)

- **children** - the array of slots collected for a specific object:
  - **label** - the name of the slot
  - **slot_id** - the ID of the slot
  - **parent_id** - the ID of the parent object
Slots are collected for CPU, RAM, HD, and Networks. For Summary, the children array contains the list of fields that provide basic information about a compute resource/backup server.

relations - the array of custom fields:
- label - the name of the custom field
- value - the value of the custom field
- id - the ID of the custom field

40.4 Add Custom Field to Hardware Info with Slots

To add a custom field to hardware info with slots (CPU, RAM, HD, and Networks), use the following request:

POST /settings/:targets/:target_id/hardware_info/custom_fields/:field/slot/:slot_id.xml

POST /settings/:targets/:target_id/hardware_info/custom_fields/:field/slot/:slot_id.json

Where:
- targets - the resource type (hypervisors or backup_servers)
- target_id - the ID of a compute resource or backup server
- field - the default field to which a custom field is added. The following default fields can be used:
  - CPU - cpu_sockets
  - RAM - memory_slots
  - HD - disks
  - Networks - nics
- slot_id - the ID of the slot to which a custom field is added

XML Request Example


JSON Request Example

Where:

- **label** - the name of the custom field
- **value** - the value that will be displayed in the custom field

### 40.5 Add Custom Field to Hardware Info without Slots

To add a custom field to hardware info without slots (Summary), use the following request:

**POST** 
/settings/:targets/:target_id/hardware_info/custom_fields/:field.xml

**POST** 
/settings/:targets/:target_id/hardware_info/custom_fields/:field.json

Where:

- **targets** - the resource type (**hypervisors** or **backup_servers**)
- **target_id** - the ID of a compute resource or backup server
- **field** - the default field to which a custom field is added. The following default fields can be used:
  - uptime
  - cpu
  - memory
  - server_type
  - os
  - manufacturer_model
  - bios_serial_number

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:

* label* - the name of the custom field

* value* - the value that will be displayed in the custom field

### 40.6 Edit Custom Field in Hardware Info with Slots

To edit a custom field added to hardware info with slots (CPU, RAM, HD, and Networks), use the following request:

**PUT**

/`settings/:targets/:target_id/hardware_info/custom_fields/:field/slot/:slot_id/:id.xml`

**PUT**

/`settings/:targets/:target_id/hardware_info/custom_fields/:field/slot/:slot_id/:id.json`

Where:

* **targets** - the resource type (hypervisors or backup_servers)
* **target_id** - the ID of a compute resource or backup server
* **field** - the default field where a custom field is edited. The following default fields can be used:
  * CPU - `cpu_sockets`
  * RAM - `memory_slots`
  * HD - `disks`
  * Networks - `nics`
* **slot_id** - the ID of the slot where a custom field is edited
* **id** - the ID of the custom field

**XML Request Example**

```shell
```

**JSON Request Example**

```shell
```
40.7 Edit Custom Field in Hardware Info without Slots

To edit a custom field added to hardware info without slots (Summary), use the following request:

PUT /settings/:targets/:target_id/hardware_info/custom_fields/:field/:id.xml

PUT /settings/:targets/:target_id/hardware_info/custom_fields/:field/:id.json

Where:

targets - the resource type (hypervisors or backup_servers)
target_id - the ID of a compute resource or backup server
field - the default field where a custom field is edited. The following default fields can be used:

- uptime
- cpu
- memory
- server_type
- os
- manufacturer_model
- bios_serial_number

id - the ID of the custom field

XML Request Example

### JSON Request Example

```bash
```

Where:

- **label** - the name of the custom field
- **value** - the value that will be displayed in the custom field

### 40.8 Delete Custom Field from Hardware Info with Slots

To delete a custom field from hardware info with slots (CPU, RAM, HD, and Networks), use the following request:

```bash
DELETE /settings/:targets/:target_id/hardware_info/custom_fields/:field[slot]/:slot_id/:id.xml
```

```bash
DELETE /settings/:targets/:target_id/hardware_info/custom_fields/:field[slot]/:slot_id/:id.json
```

Where:

- **targets** - the resource type (hypervisors or backup_servers)
- **target_id** - the ID of a compute resource or backup server
- **field** - the default field from which a custom field is deleted. The following default fields can be used:
  - CPU - cpusockets
  - RAM - memory_slots
  - HD - disks
  - Networks - nics
- **slot_id** - the ID of the slot from which a custom field is deleted
- **id** - the ID of the custom field

### XML Request Example

```bash
```

### JSON Request Example

```bash
```
Returns HTTP 204 response on successful deletion and HTTP 404 when there is no custom field with the requested ID or URL is incorrect.

40.9 Delete Custom Field from Hardware Info without Slots

To delete a custom field from hardware info without slots (Summary), use the following request:

```
```

Returns HTTP 204 response on successful deletion and HTTP 404 when there is no custom field with the requested ID or URL is incorrect.

40.9 Delete Custom Field from Hardware Info without Slots

To delete a custom field from hardware info without slots (Summary), use the following request:

```
```

Where:
- **targets** - the resource type (hypervisors or backup_servers)
- **target_id** - the ID of a compute resource or backup server
- **field** - the default field from which a custom field is deleted. The following default fields can be used:
  - `uptime`
  - `cpu`
  - `memory`
  - `server_type`
  - `os`
  - `manufacturer_model`
  - `bios_serial_number`
- **id** - the ID of the custom field

XML Request Example

```
```

JSON Request Example

```
```

Returns HTTP 204 response on successful deletion and HTTP 404 when there is no custom field with the requested ID or URL is incorrect.
OnApp High Availability brings new opportunity to deploy more than one Control Panel within one cloud. This chapter provides the API requests for the HA configuration.

Please contact your account manager to enable High Availability Control Panel for your cloud.

41.1 Get List of Clusters

To view the list of clusters, use the following request:

GET /settings/availability/clusters.xml
GET /settings/availability/clusters.json

XML Request Example

curl -i -X GET -u user:userpass -H 'Accept: application/xml'
http://onapp.test/settings/availability/clusters.xml -H 'Content-type: application/xml'

JSON Request Example

curl -i -X GET -u user:userpass -H 'Accept: application/json'
http://onapp.test/settings/availability/clusters.json 'Content-type: application/json'

XML Output Example
<availability_clusters type="array">
  <availability_cluster>
    <created_at type="datetime">2015-10-28T16:11:38+02:00</created_at>
    <id type="integer">7</id>
    <name>UI</name>
    <net_mask type="string">24</net_mask>
    <ports nil="true"/>
    <created_at type="datetime">2015-11-10T14:22+02:00</created_at>
    <virtual_ip type="string">1.1.1.1</virtual_ip>
    <nodes type="array">
      <node>
        <created_at type="datetime">2015-10-28T16:11:38+02:00</created_at>
        <id type="integer">7</id>
        <interface eth5>
          <ip_address>2.2.2.2</ip_address>
          <priority type="integer">104</priority>
        </interface>
        <created_at type="datetime">2015-10-28T16:11:38+02:00</created_at>
        <hostname>onapp2.ha.host</hostname>
      </node>
    </nodes>
  </availability_cluster>
  <availability_cluster>
  </availability_cluster>
  ...
</availability_clusters>

Where:

*created_at* - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

*id* - the ID of the high availability cluster

*name* - the label of the cluster

*net_mask* - mask of the network

*ports* - cluster ports

*state* - the state of the cluster, a cluster can have the following states:

- created - a cluster is 'created' when it is newly added, however, the changes to the HA configuration have not yet been saved
- stable - a cluster is 'stable' if it did not undergo any changes. The cluster will not be altered when the changes to the HA configuration will be saved.
- modified - a cluster is 'modified' if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a cluster will become 'stable'.
- deactivated - a cluster is 'deactivated' if the user chosen to make it inactive.

*updated_at* - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

*virtual_ip* - the virtual IP address for the cluster

*nodes* - the array of parameters for each node, where:

*created_at* - the date when the node has been created

*host_id* - the host ID

*id* - the node ID

*interface* - the network interface of the node
**ip_address** - the physical IP address of the node

**priority** - the priority for the node. The node with the highest priority will take over the virtual IP address when the component of the cluster fails.

**state** - the state of the node, a node can have the following states:

- **created** - a node is ‘created’ when it is newly added, however, the changes to the HA configuration have not yet been saved
- **stable** - a node is ‘stable’ if it did not undergo any changes. The node will not be altered when the changes to the HA configuration will be saved.
- **modified** - a node is ‘modified’ if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a node will become ‘stable’.

**updated_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**hostname** - the host name of the the host with which this node is associated

### 41.2 Get List of Cluster Nodes

To view the nodes of a cluster, use one of the following requests:

GET /settings/availability/clusters/:cluster_id.xml
GET /settings/availability/clusters/:cluster_id.json

**XML Request Example**

```bash
curl -i -X GET -u user:userpass -H 'Accept: application/xml'
http://onapp.test/settings/availability/clusters/12.xml
-H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X GET -u user:userpass -H 'Accept: application/json'
http://onapp.test/settings/availability/clusters/12.json
-H 'Content-type: application/json'
```

or

GET /settings/availability/clusters/:cluster_id/nodes.xml
GET /settings/availability/clusters/:cluster_id/nodes.json

**XML Request Example**

```bash
curl -i -X GET -u user:userpass -H 'Accept: application/xml'
http://onapp.test/settings/availability/clusters/12/nodes.xml
-H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X GET -u user:userpass -H 'Accept: application/json'
http://onapp.test/settings/availability/clusters/12/nodes.json
-H 'Content-type: application/json'
```
curl -i -X GET -u user:userpass -H 'Accept: application/json'
http://onapp.test/settings/availability/clusters/12/nodes.json 'Content-type: application/json'

XML Output Example

```
<availability_nodes type="array">
  <availability_node>
    <created_at type="datetime">2015-10-28T16:11:38+02:00</created_at>
    <host_id type="integer">2</host_id>
    <id type="integer">7</id>
    <interface>eth5</interface>
    <ip_address>2.2.2.2</ip_address>
    <priority type="integer">104</priority>
    <state>updated</state>
    <updated_at type="datetime">2015-10-28T16:11:38+02:00</updated_at>
    <hostname>onapp2.ha.host</hostname>
  </availability_node>
  ...</availability_node>
</availability_nodes>
```

Where:
- created_at - the date when the node was created
- host_id - the host ID
- id - the node ID
- interface - the network interface of the node
- ip_address - the physical IP address of the node
- priority - the priority for the node. The node with the highest priority will take over the virtual IP address when the component of the cluster fails.
- state - the state of the node, a node can have the following states:
  - created - a node is 'created' when it is newly added, however, the changes to the HA configuration have not yet been saved
  - stable - a node is 'stable' if it did not undergo any changes. The node will not be altered when the changes to the HA configuration will be saved.
  - modified - a node is 'modified' if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a node will become 'stable'.
- updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- hostname - the host name of the host with which this node is associated

### 41.3 Get Node Details

To view the details of a node, use the following request:

GET /settings/availability/clusters/:cluster_id/nodes/:node_id.xml
GET /settings/availability/clusters/:cluster_id/nodes/:node_id.xml

XML Request Example
curl -i -X GET -u user:userpass -H 'Accept: application/xml'
http://onapp.test/settings/availability/clusters/24/nodes/7.xml
-H 'Content-type: application/xml'

JSON Request Example

curl -i -X GET -u user:userpass -H 'Accept: application/json'
http://onapp.test/settings/availability/clusters/24/nodes/7.json
-H 'Content-type: application/json'

XML Output Example

<availability_node>
    <created_at type="datetime">2015-10-28T16:11:38+02:00</created_at>
    <host_id type="integer">2</host_id>
    <id type="integer">7</id>
    <interface>eth5</interface>
    <ip_address>2.2.2.2</ip_address>
    <priority type="integer">104</priority>
    <state>updated</state>
    <updated_at type="datetime">2015-10-28T16:11:38+02:00</updated_at>
    <hostname>onapp2.ha.host</hostname>
</availability_node>

Where:

- **created_at**: the date when the node has been created
- **host_id**: the host ID
- **id**: the node ID
- **interface**: the network interface of the node
- **ip_address**: the physical IP address of the node
- **priority**: the priority for the node. The node with the highest priority will take over the virtual IP address when the component of the cluster fails.
- **state**: the state of the node, a node can have the following states:
  - **created**: a node is 'created' when it is newly added, however, the changes to the HA configuration have not yet been saved
  - **stable**: a node is 'stable' if it did not undergo any changes. The node will not be altered when the changes to the HA configuration will be saved.
  - **modified**: a node is 'modified' if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a node will become 'stable'.
- **updated_at**: the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **hostname**: the host name of the host with which this node is associated

41.4 Get List of Hosts

To view the list of hosts in a cluster, use the following request:

GET /settings/availability/hosts.xml
GET /settings/availability/hosts.json

XML Request Example

curl -i -X GET -u user:userpass

JSON Request Example

curl -i -X GET -u user:userpass

XML Output Example

<availability_hosts type="array">
  <availability_host>
    <created_at type="datetime">2015-09-30T12:26:02+03:00</created_at>
    <hostname>onapp1.ha.host</hostname>
    <id type="integer">1</id>
    <state>stable</state>
    <updated_at type="datetime">2015-09-30T12:26:02+03:00</updated_at>
  </availability_host>
  <availability_host>
    <created_at type="datetime">2015-09-30T12:26:02+03:00</created_at>
    <hostname>onapp2.ha.host</hostname>
    <id type="integer">2</id>
    <state>stable</state>
    <updated_at type="datetime">2015-09-30T12:26:02+03:00</updated_at>
  </availability_host>
</availability_hosts>

Where:

created_at - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
hostname - the hostname of the host
id - ID of the host
state - the state of the host, a host can have the following states:
  - created - a host is 'created' when it is newly added, however, the changes to the HA configuration have not yet been saved.
  - stable - a host is 'stable' if it did not undergo any changes. The host will not be altered when the changes to the HA configuration will be saved.
  - modified - a host is 'modified' if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a host will become 'stable'.
updated_at - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

41.5 Get Host Nodes

To view the list of nodes within a host and their details, use the following request:
GET /settings/availability/hosts/:id.xml
GET /settings/availability/hosts/:id.json

**XML Request Example**

curl -i -X GET -u user:userpass
-H 'Accept: application/xml' -H 'Content-type:application/xml

**JSON Request Example**

curl -i -X GET -u user:userpass
-H 'Accept: application/json' -H 'Content-type:application/json

**XML Output Example**

```xml
<availability_nodes type="array">
  <availability_node>
    <created_at type="datetime">2015-10-28T16:11:38+02:00</created_at>
    <host_id type="integer">1</host_id>
    <id type="integer">9</id>
    <interface>eth0</interface>
    <ip_address>000.0.0.0</ip_address>
    <priority type="integer">0</priority>
    <state>updated</state>
    <updated_at type="datetime">2015-10-28T16:11:38+02:00</updated_at>
    <hostname>onapp1.ha.host</hostname>
  </availability_node>
  ...</availability_node>
</availability_nodes>
```

**Where:**

- **created_at** - the date when the node has been created
- **host_id** - the host ID
- **id** - the node ID
- **interface** - the network interface of the node
- **ip_address** - the physical IP address of the node
- **priority** - the priority for the node. The node with the highest priority will take over the virtual IP address when the component of the cluster fails.
- **state** - the state of the node, a node can have the following states:
  - **created** - a node is ‘created’ when it is newly added, however, the changes to the HA configuration have not yet been saved
  - **stable** - a node is ‘stable’ if it did not undergo any changes. The node will not be altered when the changes to the HA configuration will be saved.
  - **modified** - a node is ‘modified’ if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a node will become ‘stable’.
- **updated_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **hostname** - the hostname of the node
41.6 Get List of Communication Rings

To view the list of communication rings, use the following request:

GET /settings/availability/communication_rings.xml
GET /settings/availability/communication_rings.json

**XML Request Example**

```bash
curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml'
http://onapp.test/settings/availability/communication_rings.xml
```

**JSON Request Example**

```bash
curl -i -X GET -u admin:changeme -H 'Accept: application/json' -H 'Content-Type: application/json'
http://onapp.test/settings/availability/communication_rings.json
```

**XML Output Example**

```xml
<infrastructure_ha_communication_rings type="array">
  <infrastructure_ha_communication_ring>
    <bindnetaddr>000.000.000.0</bindnetaddr>
    <id>11111111111111111</id>
    <mcastaddr>000.00.0.0</mcastaddr>
    <mcastport>0000</mcastport>
    <state nil="true"/>
    <ttl>00</ttl>
  </infrastructure_ha_communication_ring>
  ...
  <infrastructure_ha_communication_ring>
</infrastructure_ha_communication_rings>
```

**Where:**

- `bindnetaddr` - the multicast network used by the hosts to communicate with each other
- `id` - the ID of the communication ring
- `mcastaddr` - the multicast IP address
- `mcastport` - the multicast port
- `state` - the state of the communication ring, a communication ring can have the following states:
  - created - a communication ring is 'created' when it is newly added, however, the changes to the HA configuration have not yet been saved
  - stable - a communication ring is 'stable' if it did not undergo any changes. The communication ring will not be altered when the changes to the HA configuration will be saved.
  - modified - a communication ring is 'modified' if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a communication ring will become 'stable'.
- `ttl` - time to live
41.7 Get Details of Communication Ring

To view the details of a particular communication ring, use the following request:

GET
/settings/availability/communication_rings/:communication_ring_id.xml
GET
/settings/availability/communication_rings/:communication_ring_id.json

XML Request Example

```
curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml'
http://onapp.test/settings/availability/communication_rings/24.xml
```

JSON Request Example

```
curl -i -X GET -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json'
http://onapp.test/settings/availability/communication_rings/24.json
```

XML Output Example

```
<communication_ring>
  <bindnetaddr>000.000.00.0</bindnetaddr>
  <id>24</id>
  <mcastaddr>000.00.0.0</mcastaddr>
  <mcastport>0000</mcastport>
  <ttl>00</ttl>
</communication_ring>
```

Where:

- **bindnetaddr** - the multicast network used by the hosts to communicate with each other
- **id** - the ID of the communication ring
- **mcastaddr** - the multicast IP address
- **mcastport** - the multicast port
- **state** - the state of the communication ring, a communication ring can have the following states:
  - created - a communication ring is 'created' when it is newly added, however, the changes to the HA configuration have not yet been saved
  - stable - a communication ring is 'stable' if it did not undergo any changes. The communication ring will not be altered when the changes to the HA configuration will be saved.
  - modified - a communication ring is 'modified' if it has been edited, however, the changes to the HA configuration have not yet been saved. When the changes will be applied, the status of such a communication ring will become 'stable'.
- **ttl** - time to live
41.8 Get Status of OnApp Subsystems

To view the status of OnApp Engine subsystems, use the following request:

GET /sysadmin_tools/infrastructure/services.xml
GET /sysadmin_tools/infrastructure/services.json

XML Request Example

```sh
```

JSON Request Example

```sh
```

XML Output Example

```
<objects type="array">
  <object>
    <node_name>onapp_ha_cp1</node_name>
    <services type="array">
      <service>
        <name>cloudboot-ip</name>
        <pid>N/A</pid>
        <status>N/A</status>
      </service>
      <service>
        <name>cron</name>
        <pid type="integer">17004</pid>
        <status>Online</status>
      </service>
      <service>
        <name>dhcpd</name>
        <pid>N/A</pid>
        <status>N/A</status>
      </service>
      ...
    </services>
  </object>
  ...
</objects>
```

The output contains a number of nodes each containing an array of services running on this node.

**Where:**

- **node_name** - node name.
- **name** - name of service/process
- **pid** - PID of related service/process. Can be a digit or "N/A"
- **status** - human readable status of service/process. Can be "online" or "offline"
41.9 Enable High Availability

To enable high availability for your cloud, use the following request:

PUT /settings/availability/enable.xml
PUT /settings/availability/enable.json

**XML Request Example**
```
curl -i -X PUT -u user:userpass
    http://onapp.test/settings/availability/enable.xml
    -H 'Accept: application/xml' -H 'Content-type: application/xml' -d ''
```

**JSON Request Example**
```
curl -i -X PUT -u user:userpass
    http://onapp.test/settings/availability/enable.json
    -H 'Accept: application/json' -H 'Content-type: application/json' -d '{}'
```

41.10 Disable High Availability

To disable high availability for your cloud, use the following request:

PUT /settings/availability/disable.xml
PUT /settings/availability/disable.json

**XML Request Example**
```
curl -i -X PUT -u user:userpass
    http://onapp.test/settings/availability/disable.xml
    -H 'Accept: application/xml' -H 'Content-type: application/xml' -d ''
```

**JSON Request Example**
```
curl -i -X PUT -u user:userpass
    http://onapp.test/settings/availability/disable.json
    -H 'Accept: application/json' -H 'Content-type: application/json' -d '{}'
```

41.11 Deactivate Cluster

To deactivate a cluster, use the following request:

PUT /settings/availability/clusters/:cluster_id/deactivate.xml
PUT /settings/availability/clusters/:cluster_id/deactivate.json

**XML Request Example**
```
curl -i -X PUT -u user:userpass
    http://onapp.test/settings/availability/clusters/:cluster_id/deactivate.xml
    -H 'Accept: application/xml' -H 'Content-type: application/xml' -d ''
```

**JSON Request Example**
```
curl -i -X PUT -u user:userpass
    http://onapp.test/settings/availability/clusters/:cluster_id/deactivate.json
    -H 'Accept: application/json' -H 'Content-type: application/json' -d '{}'
```
### 41.12 Activate Deactivated Cluster

To activate a cluster you have previously deactivated, use the following request:

**XML Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/clusters/:cluster_id/recreate.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d ''
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/clusters/:cluster_id/recreate.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d '{}'
```

Where:

- **cluster_id** - the ID of the cluster you want to deactivate.

### 41.13 Apply Changes to High Availability Configuration

To apply changes to your HA configuration, use the following request:

**XML Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/apply_changes.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d ''
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/apply_changes.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d '{}'
```

Where:

- **cluster_id** - the ID of the cluster you want to deactivate.
41.14 Apply Changes to Multicast Configuration

To apply changes to the multicast configuration, use the following request:

**XML Request Example**
```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/communication_rings/apply.xml
-H 'Accept: application/xml' -H 'Content-type:application/xml' -d ''
```

**JSON Request Example**
```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/communication_rings/apply.json
-H 'Accept: application/json' -H 'Content-type:application/json' -d '{}'
```

41.15 Edit Host

To edit a host, use the following request:

**XML Request Example**
```
curl -i -X PUT -u user:userpass
-d '<availability_host><hostname>new.ha.host</hostname></availability_host>'
-H 'Accept: application/xml' -H 'Content-type:application/xml'
```

**JSON Request Example**
```
curl -i -X PUT -u user:userpass
-d '{"availability_host":{"hostname":"onapp.ha.host"}}'
-H 'Accept:application/json' -H 'Content-type:application/json'
```

Where:
hostname - the hostname of the host

41.16 Edit Cluster

To edit a cluster, use the following request:

PUT /settings/availability/clusters/:cluster_id.xml
PUT /settings/availability/clusters/:cluster_id.json

XML Request Example

curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml' -d '
<availability_cluster><virtual_ip>1.1.1.1</virtual_ip><name>UI</name><nodes><id>1</id><hostname>ha-cp1</hostname><interface>eth5</interface><ip_address>2.2.2.2</ip_address><priority>104</priority></nodes></availability_cluster>'

JSON Request Example

curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json' -d '{"availability_cluster": {"virtual_ip": "1.1.1.1", "name": "UI", "nodes": [{"id": "1", "hostname": "ha-cp1", "interface": "eth5", "ip_address": "2.2.2.2", "priority": "104"}]}}'

Where:

availability_cluster - the array of the high availability cluster parameters

virtual_ip - set the desired virtual IP address of the control panels

name - the cluster type (UI for UI cluster, Daemon for Daemon cluster, Cloud_boot for Cloudboot cluster, Load_balancer for load balancer cluster, Redis for Redis cluster and Message_queue for message queue cluster)

nodes - the array of the node's parameters

id - the ID of the node

hostname - the hostname of the node

interface - set the network interface for the node

ip_address - the IP address of the node

priority - set the priority for the node. The node with the highest priority will take over the virtual IP address when the component of the cluster fails.

41.17 Edit Node

To edit a node, use the following request:

PUT /settings/availability/clusters/:cluster_id/nodes/:node_id.xml
PUT /settings/availability/clusters/:cluster_id/nodes/:node_id.json

XML Request Example
curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml' -d
'<availability_node><host_id>5</host_id><interface>eth6</interface><ip_address>2.2.3.2</ip_address><priority>104</priority></availability_node>'

JSON Request Example

```
curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json' -d '{"availability_node": { "host_id": "5", "interface": "eth6", "ip_address": "2.2.3.2", "priority": "104" } }'
```

Where:

- **host_id** - the host ID
- **interface** - set the network interface of the node
- **ip_address** - the IP address of the node
- **priority** - set the priority for the node. The node with the highest priority will take over the virtual IP address when the component of the cluster fails.

### 41.18 Edit Communication Ring

To edit a communication ring, use the following request:

PUT
/settings/availability/communication_rings/:communication_ring_id.xml

PUT
/settings/availability/communication_rings/:communication_ring_id.json

**XML Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/communication_rings/12.xml -H
'Accept: application/xml' -H 'Content-type:application/xml' -d
'<ring><bindnetaddr>000.000.00.0</bindnetaddr><mcastaddr>000.00.00.0</mcastaddr><mcastport>0000</mcastport><ttl>00</ttl></ring>'
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass
http://onapp.test/settings/availability/communication_rings/12.json -H
'Accept: application/json' -H 'Content-type:application/json' -d
'{"ring":{"bindnetaddr":"000.000.00.0","mcastaddr":"000.00.00.0","mcastport":"0000","ttl":"00"}}'
```

Where:

- **bindnetaddr** - the multicast network used by the hosts to communicate with each other
- **mcastaddr** - the multicast IP address
- **mcastport** - the multicast port
**41.19 Add Cluster**

To create a cluster, use the following request:

POST /settings/availability/clusters.xml

POST /settings/availability/clusters.json

XML Request Example

```
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<availability_cluster><name>MQ</name><virtual_ip>100.0.100.100</virtual_ip></availability_cluster>' --url http://onapp.test/settings/availability/clusters.xml
```

JSON Request Example

```
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"availability_cluster":{"name":"MQ","virtual_ip":"100.0.100.100"}}' --url http://onapp.test/settings/availability/clusters.json
```

Where:

- **name** - the label of the new cluster. The name of the cluster should be unique and should be one of the following: DAEMON, UI, CLOUD_BOOT, LB, DB, REDIS, MQ
- **virtual_ip** - the virtual IP address of the cluster. This IP address should be unique.

**41.20 Add Host**

To add a new host, use the following request:

POST /settings/availability/hosts.xml

POST /settings/availability/hosts.json

XML Request Example

```
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<availability_host><hostname>new.ha.host</hostname></availability_host>'
```

JSON Request Example

```
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"availability_host":{"hostname":"new.ha.host"}}'
```

Where:
hostname* - the hostname of the host. This hostname should be unique.

41.21 Add New Node to Cluster

To add a node to cluster, use the following request:
POST /settings/availability/clusters/:cluster_id/nodes.xml
POST /settings/availability/clusters/:cluster_id/nodes.json

**XML Request Example**

curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'<availability_node><host_id>8</host_id><interface>eth5</interface><ip_address>2.2.2.2</ip_address><priority>104</priority></availability_node>' --url http://onapp.test/settings/availability/clusters/18/nodes.xml

**JSON Request Example**

curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{ "availability_node": { "host_id": "8", "interface": "eth5", "ip_address": "2.2.2.2", "priority": "104" } }' --url http://onapp.test/settings/availability/clusters/18/nodes.json

Where:

host_id* - the host ID. This ID should be a unique within a cluster.
interface* - the network interface of the node
ip_address* - the physical IP address of the node
priority* - the priority for the node. The node with the highest priority will take over the virtual IP address when the component of the cluster fails.

41.22 Add Communication Interface

To create a new communication ring, use the following request:
POST /settings/availability/communication_rings.xml
POST /settings/availability/communication_rings.json

**XML Request Example**

curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'(<ring><bindnetaddr>000.000.00.0</bindnetaddr><mcastaddr>000.00.00.00</mcastaddr><mcastport>0000</mcastport><ttl>00</ttl></ring>)'

**JSON Request Example**

curl -i -X POST -u user:userpass
'(<ring><bindnetaddr>000.000.00.0</bindnetaddr><mcastaddr>000.00.00.00</mcastaddr><mcastport>0000</mcastport><ttl>00</ttl></ring>)'
curl -i -X POST -u user:userpass
http://onapp.test/settings/availability/communication_rings.json -H 'Accept: application/json' -H 'Content-type:application/json' -d '{"ring":{"bindnetaddr":"000.000.00.0","mcastaddr":"000.00.00.00","mcastport":"0000","ttl":"00"}}'

Where:

bindnetaddr* - the multicast network used by the hosts to communicate with each other. This should be a unique network.
mcastaddr* - the multicast IP address. This should be a unique address.
mcastport* - the multicast port. This should be a unique port.
ttl* - time to live

41.23 Delete Host

To delete a host, use the following request:
DELETE /settings/availability/hosts/:host_id.xml
DELETE /settings/availability/hosts/:host_id.xml

XML Request Example

curl -i -X DELETE -u user:userpass

JSON Request Example

curl -i -X DELETE -u user:userpass

41.24 Delete Node

To delete a node, use the following request:
DELETE /settings/availability/clusters/:cluster_id/nodes/:node_id.xml
DELETE /settings/availability/clusters/:cluster_id/nodes/:node_id.json

You cannot delete a cluster node if the cluster to which this node is assigned has only two nodes. The minimum number of nodes in a cluster is 2.

XML Request Example
41.25 Delete Communication Ring

To delete a communication ring, use the following request:

DELETE /settings/availability/communication_rings/:communication_ring_id.xml

DELETE /settings/availability/communication_rings/:communication_ring_id.json

**XML Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/settings/availability/communication_rings/12.xml
-H 'Accept:application/xml' -H 'Content-type:application/xml'
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/settings/availability/communication_rings/12.json
-H 'Accept:application/json' -H 'Content-type:application/json'
```
42 Instance Packages

Instance packages are preconfigured CPU/RAM/Disk/Bandwidth packages that can be selected during the VS creation process. You can add multiple instance packages specifying different values for the parameters to suit your customer's needs. Resources that are not set when creating an instance package, such as, for example, swap disk size, are calculated automatically.

To provide your users with ability to choose VSs resources from the predefined instance package(s), add the necessary packages to the users' bucket(s). After that, instance packages will appear in the server creation wizard, on the Resources step.

42.1 Get List of Instance Packages

To view the list of instance packages, use the following request:

GET /instance_packages.xml
GET /instance_packages.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<instance_packages type="array">
    <instance_package>
        <bandwidth type="integer">1</bandwidth>
        <cpus type="integer">2</cpus>
        <created_at type="datetime">2015-06-17T18:29:26+03:00</created_at>
        <disk_size type="integer">1</disk_size>
        <id type="integer">1</id>
        <label>Test</label>
        <memory type="integer">128</memory>
        <updated_at type="datetime">2015-06-17T18:29:26+03:00</updated_at>
    </instance_package>
    ...
</instance_packages>
```

Where:

- **bandwidth** - the bandwidth available in this instance package
- **cpus** - the number of CPU cores available in this instance package
- **created_at** - time when the instance package was created, in [YYYY][MM][DD][T][hh][mm][ss]Z
- **disk_size** - the disk size available in this instance package
- **id** - ID of the instance package
**label** - the name of the instance package

**memory** - the RAM size (GB) available in the instance package

**updated_at** - time when the instance package was updated, in [YYYY][MM][DD]T[hh][mm][ss]Z

### Page History

v5.2

- removed the deprecated request method - GET /instance_types.

### 42.2 Get Instance Package Details

To view the details of an instance package, use the following request:

GET /instance_packages/:instance_package_id.xml

GET /instance_packages/:instance_package_id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```xml
<instance_package>
  <id type="integer">1</id>
  <label>Test</label>
  <cpus type="integer">2</cpus>
  <disk_size type="integer">1</disk_size>
  <memory type="integer">128</memory>
  <bandwidth type="integer">1</bandwidth>
  <created_at type="datetime">2015-06-17T15:26:00+00:00</created_at>
  <updated_at type="datetime">2015-06-17T15:26:00+00:00</updated_at>
  <billing_plan_ids type="array">
    <fixnum type="integer">1</fixnum>
    <fixnum type="integer">5</fixnum>
    <fixnum type="integer">140</fixnum>
  </billing_plan_ids>
</instance_package>
```

**Where:**

- **id** - ID of the instance package
- **label** - the name of the instance package
- **cpus** - the number of CPU cores available in this instance package
- **disk_size** - the disk size available in this instance package
- **memory** - the RAM size (GB) available in the instance package
bandwidth - the bandwidth available in this instance package

created_at - time when the instance package was created, in [YYYY][MM][DD][hh][mm][ss]Z

updated_at - time when the instance package was updated, in [YYYY][MM][DD][hh][mm][ss]Z

billing_plan_ids - the array of billing plans to which this instance package was added

fixnum - the ID of the billing plan in which the instance package was used

Page History

v.5.2

- removed the deprecated request method -

GET /instance_types/:instance_type_id.

42.3 Add Instance Package

To add a new instance package, use the following request:

POST /instance_packages.xml

POST /instance_packages.json

XML Request Example

curl -i -X POST -d 'xml Request Example

<instance_package><label>example</label><cpus>1</cpus><memory>1024</memory><disk_size>50</disk_size><bandwidth>100</bandwidth></instance_package>'


JSON Request Example

curl -i -X POST -d '{"instance_package": {"label":"example", "cpus":"1", "memory":"1024", "disk_size":"50", "bandwidth":"100"}}' -u user:userpass http://onapp.test/instance_packages.json -H 'Accept: application/json' -H 'Content-type: application/json'

Where:

label - the name of the instance package

cpus - the number of CPU cores available in this instance package. The maximum CPUs value is 8.

memory - the RAM size (GB) available in the instance package. The maximum value is 16384 MB by default.

disk_size - the disk size available in this instance package. The maximum value is 100 GB by default.

bandwidth - the bandwidth available in this instance package. Set this parameter to '0' to make bandwidth in this instance package unlimited.

Bandwidth calculation is based on max_network_interface_port_speed configuration parameter in on_app.yml file. Example:

If you have max port speed equal to 2000 Mbit/second, bandwidth could not be more than 2000*3600(seconds in one hour)/(1000*8)=900 GB per hour.
You can change the default minimum and/or maximum values for memory, disk size and bandwidth by adding the following parameters to the config/on_app.yml file and restarting OnApp services:

- `instance_package_min_disk_size` (GB)
- `instance_package_max_disk_size` (GB)
- `instance_package_max_memory` (MB)
- `instance_package_min_bandwidth` (GB)

Page History

v5.2

- removed the deprecated request method - POST /instance_types.

42.4 Edit Instance Package

To edit an instance package, use the following request:

**PUT** /instance_packages/:instance_package_id.xml

**PUT** /instance_packages/:instance_package_id.json

**XML Request Example**

```
curl -i -X PUT -d '<instance_package><label>new_label</label><cpus>1</cpus><memory>1024</memory><disk_size>50</disk_size><bandwidth>100</bandwidth></instance_package>' -u user:userpass http://onapp.test/instance_packages/1.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X PUT -d '{"instance_package": {"label":"new_label", "cpus":"1", "memory":"1024", "disk_size":"50", "bandwidth":"100"}}' -u user:userpass http://onapp.test/instance_packages/1.json -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- `label` - the name of the instance package
- `cpus` - the number of CPU cores available in this instance package. The maximum CPUs value is 8.
- `memory` - the RAM size available in the instance package. The maximum value is 16384 MB by default.
- `disk_size` - the disk size available in this instance package. The maximum value is 100 GB by default.
bandwidth - the bandwidth available in this instance package. Set this parameter to '0' to make bandwidth in this instance package unlimited.

You can change the default minimum and/or maximum values for memory, disk size and bandwidth by adding the following parameters to the config/on_app.yml file and restarting OnApp services:

- instance_package_min_disk_size (GB)
- instance_package_max_disk_size (GB)
- instance_package_max_memory (MB)
- instance_package_min_bandwidth (GB)

Page History
v5.2
- removed the deprecated request method - PUT /instance_types/:instance_type_id.

42.5 Delete Instance Package

To delete an instance package, use the following request:

DELETE /instance_packages/:instance_package_id.xml
DELETE /instance_packages/:instance_package_id.xml

XML Request Example


JSON Request Example


Page History
v.5.2
- removed the deprecated request method - DELETE /instance_types/:instance_type_id.
43 Integrated Storage

Integrated Storage functionality allows to build a highly scalable and resilient SAN using local disks in compute resource. Using the Integrated Storage, you can create a virtual data store in OnApp Cloud that spans multiple physical drives in compute resources, with RAID-like replication and striping across drives.

43.1 Get List of Integrated Storage Data Stores

To get the list of integrated storage data stores, use the following request:

GET /storage/:hvz_id/data_stores.xml
GET /storage/:hvz_id/data_stores.json

XML Request Example


JSON Request Example


Where:

hvz_id - compute zone ID (storage API enpoint zone)

XML Output Example

```xml
<data_stores type="array">
  <data_store>
    <id>7fzvrpeyw46j3a</id>
    <name>igor-datastore-3</name>
    <total_size type="integer">160041885696</total_size>
    <free_size type="integer">160016719872</free_size>
    <performance type="integer">0</performance>
    <disk_count type="integer">2</disk_count>
  </data_store>
</data_stores>
```

Where:

data_stores - an array of data stores with the following details

id - integrated data store ID
name - data store label
total_size - the total size of the integrated data store in bytes
free_size - the free size of the integrated data store in bytes

performance - the storage node performance level (0 - Low, 1 - Normal, 2 - High performance)
disk_count - the number of disks on this data store
nodes - an array of nodes with their details
id - node ID

43.2 Get Integrated Data Store Details

To view details of a particular integrated data store, use the following request:

GET /storage/:hvz_id/data_stores/:data_store_id.xml
GET /storage/:hvz_id/data_stores/:data_store_id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

Where:

hvz_id - compute zone ID (storage API endpoint)
data_store_id - data store ID

XML Output Example

```xml
<data_store>
  <id>7fzvrpeyw46j3a</id>
  <name>igor-datastore-3</name>
  <total_size type="integer">160041885696</total_size>
  <free_size type="integer">160016719872</free_size>
  <performance type="integer">0</performance>
  <disk_count type="integer">2</disk_count>
  <nodes type="array">
    <node>
      <id>3233314811</id>
    </node>
  </nodes>
</data_store>
```

Where:

id - integrated data store ID
name - data store label
total_size - the total size of the integrated data store in bytes
**free_size** - the free size of the integrated data store in bytes

**performance** - the storage node performance level (0: Low, 1: Normal, 2: High performance)

**disk_count** - the number of disks on this data store

**nodes** - an array of nodes with their details

**id** - node ID

### 43.3 Add Integrated Storage Data Store

To add a new integrated storage data store, use the following request:

**POST** /storage/:hvz_id/data_stores.xml

**POST** /storage/:hvz_id/data_stores.json

**XML Request Example**

```
curl -X POST -d '<storage_data_store><name>datastore-test-2</name><replicas>1</replicas><stripes>1</stripes><node_ids type="array"><string>190496273</string></node_ids><overcommit>0</overcommit></storage_data_store>' http://onapp.test/storage/12/data_stores.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass
```

**JSON Request Example**

```
curl -X POST -d '{"storage_data_store":{"name":"datastore-test-2","replicas":"1","stripes":"1","overcommit":"0","node_ids":["125120933","663678591"]}}' http://onapp.test/storage/12/data_stores.json -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass
```

Where:

**hvz_id** - compute zone ID (storage API endpoint zone)

**replicas** - the number of replicas that will be used for data in this datastore and could be in range 1, 2, 4. The default value is 2.

**stripes** – the number of stripes the data will be divided into. It can also be 1, 2, 4. The default value is 1.

**node_ids** – the array of identifiers of storage nodes, that will be used to store the data.

**overcommit** - overcommit percentage. You can set overcommit to 0, 20, 50 or 100. The default value is 0.

**JSON Output Example:**

```
{"name":"datastore-igor-2","replicas":"1","stripes":"1","performance":"0","overcommit":"0","owner_ids":["125120933","663678591"]},"owners":["125120933,663678591","members":"125120933,663678591","members ship_count":2,"uuid":"51j3ac7hszbgro","provisioning":0,"result":"SUCCESS"}
```

Where:
**performance** - the storage node performance level (0 - Low, 1 - Normal, 2 - High performance)

**owner_ids** - the array of identifiers of storage nodes, that are used to store data

**owners** - the array of identifiers of storage nodes, that are used to store data

**members** - the storage node ids that are used to store the data

**membership_count** - the total number of storage nodes used

**uuid** - unique datastore identifier.

**Page History**

v. 3.2
- added *hvz_id* parameter

v. 3.0
- added *overcommit* parameter

### 43.4 Edit Integrated Data Store

To edit the integrated data store, use the following request:

```plaintext
PUT /storage/:hvz_id/data_stores/:id.xml
PUT /storage/:hvz_id/data_stores/:id.json
```

Where:

- **hvz_id** - compute zone ID (storage API endpoint zone)
- **id** - data store ID

**XML Request Example**

```bash
curl -X PUT -d '<?xml version="1.0" encoding="UTF-8"?>
<owner_ids type="array">
  <string>190496273</string>
</owner_ids>'
http://onapp.test/storage/2/data_stores/12.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass
```

**JSON Request Example**

```bash
curl -X PUT -d '{"data_store": {"owner_ids": ["125120933", "663678591"]}}'
http://onapp.test/storage/2/data_stores/12.json
-H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass
```

Where:

**owner_ids** – the array of identifiers of storage nodes, that will be used to store the data.

You will get a 204 status response on success, and 404 if there is no such data store with a requested ID or you entered incorrect URL.

**Page History**

v. 3.2
- added *hvz_id* parameter
43.5 Delete Integrated Storage Data Store

To delete an integrated storage data store, use the following request:

DELETE /storage/:hvz_id/data_stores/:id.xml
DELETE /storage/:hvz_id/data_stores/:id.json

XML Request Example


JSON Request Example


Where:

hvz_id - compute zone ID (storage API endpoint zone)
id - data store ID

You will get a 204 status response on success, and 404 if there is no such data store with a requested ID or you entered incorrect URL.

Page History

v. 3.2

• added hvz_id parameter

43.6 Get List of Integrated Storage Data Stores Disk Drives

To get the list of integrated storage data store disk drives, use the following request:

GET /storage/:hvz_id/data_stores/:data_store_id/disks.xml
GET /storage/:hvz_id/data_stores/:data_store_id/disks.json

XML Request Example


JSON Request Example


Where:

hvz_id - compute zone ID (storage API endpoint zone)
data_store_id - the data store ID

**XML Output Example**

```xml
<disks type="array">
  <disk>
    <id>7fzvrpeyw6j3a</id>
  </disk>  
  <disk>
    <id>clzhym8rod9kjr</id>
  </disk>  
</disks>
```

Where:

- **disks** - an array of disks with the following details
- **id** - integrated data store ID

### 43.7 Get Integrated Storage Data Store Disk Drive Details

To view details of a particular VDisk, use the following request:

GET `/storage/:hypervisor_group_id/data_stores/:storage_data_store_id/disks/:storage_disk_id.xml`

GET `/storage/:hypervisor_group_id/data_stores/:storage_data_store_id/disks/:storage_disk_id.json`

**XML Request Example**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u admin:'password' --url http://onapp.test/storage/12/data_stores/2/disks/a1c5so6um4w0dn.xml
```

**JSON Request Example**

```
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u admin:'password' --url http://onapp.test/storage/12/data_stores/2/disks/a1c5so6um4w0dn.json
```

**XML Output Example**
<disk>
    <id>a1c5so6um4w0dn</id>
    <nodes type="array">
        <node>
            <id>16693618</id>
            <hypervisor_id type="integer">2</hypervisor_id>
        </node>
    </nodes>
</disk>

Where:
- id - the ID of the VDisk
- nodes - an array of nodes with the following details:
  - id - the ID of the node
  - hypervisor_id - the ID of the compute resource

43.8 Add Disk Drive to Integrated Storage Data Store

To add a new VDisk to the integrated storage data store, use the following request:
POST /storage/:hvz_id/data_stores/:data_store_id/disks.xml
POST /storage/:hvz_id/data_stores/:data_store_id/disks.json

XML Request Example

```
curl -X POST -d '
    '<storage_disk><name>testdisk</name><size>12</size></storage_disk>'
```

JSON Request Example

```
curl -X POST -d '{"storage_disk":{"name":"testdisk", "size":"12"}}'
```

Where you have to specify disk label and disk size in megabytes.

PLEASE NOTE: You cannot decrease size of Integrated Storage data store disks.

Page History

v. 3.2

- added hvz_id parameter
43.9 Remove Disk Drive from Integrated Storage Data Store

To delete VDisk from the integrated storage data store, use the following request:

DELETE /storage/:hvz_id/data_stores/:data_store_id/disks/:disk_id.xml
DELETE /storage/:hvz_id/data_stores/:data_store_id/disks/:disk_id.json

**XML Request Example**

```shell
curl -X DELETE http://onapp.test/storage/12/data_stores/23/disks/1.xml
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
-u user:userpass
```

**JSON Request Example**

```shell
curl -X DELETE http://onapp.test/storage/12/data_stores/23/disks/1.json
-H 'Accept: application/json'
-H 'Content-type: application/json'
-u user:userpass
```

**Where:**

**hvz_id** - the ID of the compute resource

**data_store_id** - the ID of the data store to which the disk is attached

**disk_id** - the ID of the disk you wish to remove

You will get a 204 status response on success, and 404 if there is no such data store or disk with a requested ID or you entered incorrect URL.

**Page History**

v. 3.2
- added hvz_id parameter

43.10 Get Storage Node IO Statistics

To get the storage node IO statistics, use the following request:

GET /storage/:hvz_id/nodes/:node_id/io_stats.xml
GET /storage/:hvz_id/nodes/:node_id/io_stats.json

**XML Request Example**

```shell
curl -I -X GET -u 'user:userpass'
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

**JSON Request Example**
curl -i -X GET -u 'user:userpass'

Where:

hvz_id - compute zone ID (storage API enpoint zone)
node_id - storage node ID
filter[start] - defines the start period for the stats
filter[finish] - defines the end period for which the stats should be generated
filter[type] - specifies the type of stats. Use the following mapping:
  • 0 - number of read IOs processed
  • 1 - number of read IOs merged with in-queue IO
  • 2 - number of sectors read
  • 3 - total wait time for read requests
  • 4 - number of write IOs processed
  • 5 - number of write IOs merged with in-queue IO
  • 6 - number of sectors written
  • 7 - total wait time for write requests
  • 8 - number of IOs currently in flight
  • 9 - total time this block device has been active
  • 10 - total wait time for all requests

XML Output Example

<node_io_stats type="array">
  <node_io_stat>
    <created_at type="datetime">2012-12-27T08:30:28+00:00</created_at>
    <value type="integer">0</value>
    <node_id>"1635592966"</node_id>
  </node_io_stat>
  <node_io_stat>
    <created_at type="datetime">2012-12-27T08:31:28+00:00</created_at>
    <value type="integer">0</value>
    <node_id>"1635592967"</node_id>
  </node_io_stat>
  ...
</node_io_stats>

43.11 Get Integrated Storage Datastore Disk IO Statistics

To get the Integrated Storage datastore disk IO statistics, use the following request:
GET /storage/:hvz_id/nodes/:node_id/vdisk_id/io_stats.xml
GET /storage/:hvz_id/nodes/:node_id/vdisk_id/io_stats.json

XML Request Example

JSON Request Example


Where:

hvz_id - compute zone ID (storage API endpoint zone)
node_id - storage node ID
vdisk_id - virtual disk ID

filter[start] - defines the start period for the stats
filter[finish] - defines the end period for which the stats should be generated
filter[type] - specifies the type of stats. Use the following mapping:

- 0 - number of read IOs processed
- 1 - number of read IOs merged with in-queue IO
- 2 - number of sectors read
- 3 - total wait time for read requests
- 4 - number of write IOs processed
- 5 - number of write IOs merged with in-queue IO
- 6 - number of sectors written
- 7 - total wait time for write requests
- 8 - number of IOs currently in flight
- 9 - total time this block device has been active
- 10 - total wait time for all requests

43.12 Forget Storage Node

To forget the node that is down is still visible in nodes list, use the following request:

POST /storage/nodes/:node_id/forget.xml
POST /storage/nodes/:node_id/forget.json

XML Request Example
43.13 Get List of Backend Nodes on Integrated Storage Data Store

To get the list of integrated storage backend nodes, use the following request:

GET /storage/:hvz_id/hypervisors.xml
GET /storage/:hvz_id/hypervisors.json

XML Request Example

```bash
```

JSON Request Example

```bash
curl -i -X POST -u user:userpass http://onapp.test/storage/nodes/1/forget.json?storage_endpoint_zone=71
```

Where you have to specify storage node ID and the storage API endpoint zone ID in the URL. Returns 200 status on success and 422 on failure.

XML Output Example

```xml
<hypervisor>
  <id>3</id>
  <nodes type="array">
    <node>
      <id>2</id>
    </node>
    <node>
      <id>12</id>
    </node>
  </nodes>
</hypervisor>
```

Where:

- `hvz_id` - compute zone ID (storage API endpoint zone)

XML Output Example

```xml
<hypervisor>
  <id>3</id>
  <nodes type="array">
    <node>
      <id>2</id>
    </node>
    <node>
      <id>12</id>
    </node>
  </nodes>
</hypervisor>
```

Where:
nodes - an array of nodes with their IDs

id - compute resource ID
44 IP Addresses

This class represents all the IP addresses in your installation. Use the following methods to edit, create new and delete an existing IP addresses in your cloud.

44.1 Get List of IP Addresses in IP Range

To view the list of the IP addresses in an IP range, use the following request:

GET
/settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges/:ip_range_id/ip_addresses.xml

GET
/settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges/:ip_range_id/ip_addresses.json

This API request returns only those IP addresses that are assigned to a VS and/or a user.

XML Request Example

curl -i -X GET -u user:userpass --url

JSON Request Example

curl -i -X GET -u user:userpass --url

XML Output Example
<ip_addresses type="array">
  <ip_address>
    <id type="integer">1215</id>
    <address>169.0.0.2</address>
    <prefix type="integer">24</prefix>
    <broadcast>169.0.0.255</broadcast>
    <network_address>169.0.0.0</network_address>
    <gateway>169.0.0.1</gateway>
    <created_at type="dateTime">2017-03-31T09:41:56+03:00</created_at>
    <updated_at type="dateTime">2017-03-31T09:41:56+03:00</updated_at>
    <network_id>24</network_id>
    <disallowed_primary type="boolean">false</disallowed_primary>
    <ipv4 type="boolean">true</ipv4>
    <user_id nil="true"/>
    <hypervisor_id nil="true"/>
    <ip_address_pool_id nil="true"/>
    <ip_range_id type="integer">46</ip_range_id>
  </ip_address>
  ...</ip_address>
</ip_addresses>

Where:

- **id** - the ID of the IP address
- **address** – IP address
- **prefix** - the prefix of the IP address
- **broadcast** – broadcast address
- **network_address** – the address of the network
- **gateway** – gateway address
- **created_at** — the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** — the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **network_id** – the ID of the network
- **disallowed_primary** – true if not allowed to be used as primary, otherwise false
- **customer_network_id** - the ID of the customer VLAN the IP address belongs to
- **ipv4** - whether this is an IPv4 or an IPv6 IP address: `true` for IPv4 IPs and `false` for IPv6 IPs
- **user_id** - the ID of a user the IP address is associated with
- **hypervisor_id** - the ID of a compute resource the IP address is associated with
- **ip_address_pool_id** - ID of the IP address pool the IP address is associated with
- **ip_range_id** - ID of the IP range the IP address is associated with

### 44.2 Assign IP Address to User

To assign an IP Address or several IP addresses to a particular user, so that they could create a VS based on it, use the following request:

**POST /settings/networks/:network_id/ip_addresses/assign.xml**

**POST /settings/networks/:network_id/ip_addresses/assign.json**

**XML Request Example**
**44.3 Unassign IP Address from User**

To unassign IP addresses from any user, use the following request:

**POST /settings/networks/:network_id/ip_addresses/unassign.xml**

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**Where:**

- **unassign** - the array that contains an IP address
ip_address - the IP address that you want to unassign

Page History

v 6.0
- added the unassign array

v 5.4
- the ip_address parameter accepts the full IP address. Previously this parameter accepted the ID of the IP address.

44.4 Get List of IP Addresses for VS

To get the list of IP address assignments for a particular VS, use the following request:

GET /virtual_machines/:virtual_machine_id/ip_addresses.xml
GET /virtual_machines/:virtual_machine_id/ip_addresses.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example
<ip_address_joins type="array">
  <ip_address_join>
    <id type="integer">323</id>
    <ip_address_id type="integer">5525</ip_address_id>
    <network_interface_id type="integer">267</network_interface_id>
    <created_at type="datetime">2011-07-19T12:29:10Z</created_at>
    <updated_at type="datetime">2011-07-19T12:29:10Z</updated_at>
  </ip_address_join>
  ...
</ip_address_joins>

Where:

- **ip_address_joins** – an array of all IP addresses, assigned to VS
- **id** – the ID of the IP address join
- **ip_address_id** – ID of an IP address
- **created_at** – the date when the record was created in the [YYYY][MM][DD]T[hh][mm][ss] format
- **updated_at** – the date when the record was updated in the [YYYY][MM][DD]T[hh][mm][ss] format
- **ip_address** – an array of an IP address assigned to VS
  - **id** – the ID of the IP address
  - **address** – the IP address
  - **prefix** – the prefix of the IP address
  - **broadcast** – the broadcast address
  - **network_address** – the address of the network
  - **gateway** – the gateway address
  - **created_at** – the date when the record was created in the [YYYY][MM][DD]T[hh][mm][ss] format
  - **updated_at** – the date when the record was updated in the [YYYY][MM][DD]T[hh][mm][ss] format
  - **ipv4** – whether this is an IPv4 or and IPv6 IP address: `true` for IPv4 IPs and `false` for IPv6 IPs
  - **user_id** – the ID of a user the IP address is associated with
  - **pxe** – true, if this address can be used for cloudbooting a compute resource
  - **hypervisor_id** – the ID of a compute resource the IP address is associated with
ip_range_id – the ID of the IP range the IP address is associated with
network_id – the ID of the network
ip_net_id – the ID of the IP net the IP address is associated with

44.5 Assign IP Address to VS

To assign an IP address to a virtual server, use the following request:

POST /virtual_machines/:virtual_machine_id/ip_addresses.xml
POST /virtual_machines/:virtual_machine_id/ip_addresses.json

XML Request Example

curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<ip_address><address>5.1.1.10</address><network_interface_id>84</network_interface_id><ip_net_id>147</ip_net_id><ip_range_id>210</ip_range_id><used_ip>1</used_ip></ip_address>' --url http://onapp.test/virtual_machines/12/ip_addresses.xml

JSON Request Example


Where:

ip_address - the array of parameters to allocate an IP address to VS
address - enter an IP address that you want to assign to VS
network_interface_id* - specify an ID of a network interface the IP address should be assigned to
ip_net_id - specify an ID of the IP net from which the IP address should be assigned
ip_range_id - specify an ID of the IP range from which the IP address should be assigned
used_ip - set to "1" to assign a used IP address, otherwise, set to "0"
own_ip - set to "1" to assign an IP address assigned to the current user, otherwise, set to "0"
ip_version - the version of the protocol that can be set to "6" for IPv6 or to "4" for IPv4

If you set an IPv6 or IPv4 version, the IP address is allocated automatically so you can skip the address parameter in the request.

Page History

v.5.10

• added the ip_net_id and ip_range_id parameters
v. 5.9
- added the `ip_version` parameter to automatically allocate an IPv6 or IPv4 address

v. 5.4
- the `ip_address_id` parameter has been changed to `address` and it now accepts the full IP address

### 44.6 Unassign IP Address from VS

To delete an IP address assignment from a particular VS, you have to remove the IP and rebuild the network. There are two API calls for IP deletion: one unassigns an address, but actually leaves it on a VS, and the second removes the address, rebuilds the network and thus reboots a VS.

To delete an IP address without rebuilding a network, use the following request:

```
DELETE /virtual_machines/:virtual_machine_id/ip_addresses/:id.xml
```
```
DELETE /virtual_machines/:virtual_machine_id/ip_addresses/:id.json
```

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass -u
http://onapp.test/virtual_machines/12/ip_addresses/2.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass -u
http://onapp.test/virtual_machines/12/ip_addresses/2.json
```

To delete an IP address and rebuild a network, use the following request:

```
DELETE /virtual_machines/:virtual_machine_id/ip_addresses/:id.xml?rebuild_network=1
```
```
DELETE /virtual_machines/:virtual_machine_id/ip_addresses/:id.json?rebuild_network=1
```

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass -u
http://onapp.test/virtual_machines/12/ip_addresses/2.xml?rebuild_network=1
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass -u
http://onapp.test/virtual_machines/12/ip_addresses/2.json?rebuild_network=1
```

You will get a 204 status response on success, and 404 if there is no such IP address join with a requested ID or you entered incorrect URL.
44.7 External IP Addresses

External and local IP addresses both serve the same purpose, the difference is scope. An external or public IP address is used across the entire Internet to locate computer systems and devices. A local or internal IP address is used inside a private network to locate the computers and devices connected to it. This chapter provides requests for managing external IP addresses.

44.7.1 Get External IP Address Details

To view the details of an external IP address, use the following request:

GET /ip_addresses/:ip_address_id/external_address.xml
GET /ip_addresses/:ip_address_id/external_address.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

44.7.2 Add/Edit External IP Address

To add/edit an external IP address, use the following request:

PUT /ip_addresses/:ip_address_id/external_address.xml
PUT /ip_addresses/:ip_address_id/external_address.json

ip_address_id value can be retrieved from Get List of IP Addresses for VS API request.

XML Request Example

```bash
```

JSON Request Example
Where:

- `ip_address_id` - the ID of the internal IP address
- `external_address` - set the `external_address` to show the public IP to the end user

### 44.7.3 Delete External IP Address

To remove an external IP address, use the following request:

```bash
DELETE /ip_addresses/:ip_address_id/external_address.xml
DELETE /ip_addresses/:ip_address_id/external_address.json
```

**XML Request Example**

```bash
curl -X PUT -u user:userpass --url
http://onapp.test/ip_addresses/98/external_address.xml
- H 'Accept: application/xml' -H 'Content-type: application/xml' -d
  '<?external_address>null'></external_address>'
```

**JSON Request Example**

```bash
curl -X PUT -u user:userpass --url
http://onapp.test/ip_addresses/98/external_address.json
- H 'Accept: application/json' -H 'Content-type: application/json' -d
  '{"external_address":null}''
```

### 44.8 Get IP Usage Report

To get the IP Usage report, use the following request:

```bash
GET/sysadmin_tools/ip_history/reports.csv
```

**CSV Request Example**

```bash
curl -i -X GET -u user:userpass --url
http://onapp.test/sysadmin_tools/ip_history/reports.csv -d '{"ip_address":
  "69.111.234.56", "start_date": "2019-05-20 13:57:24", "end_date": "2019-
  06-20 14:57:24"}' -o 69_168_237_52_20190520_20190620.csv
```

**CSV Output Example**

```bash
```

575
<table>
<thead>
<tr>
<th>Date/Time</th>
<th>ip_address</th>
<th>full_name</th>
<th>login</th>
<th>email</th>
<th>hostname</th>
<th>virtual_machine_identifier</th>
<th>network_interface_identifier</th>
<th>mac_address</th>
</tr>
</thead>
<tbody>
<tr>
<td>2019-06-07 08:12:44 +0300</td>
<td>00:16:3e:69:15</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:69:15</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
<tr>
<td>2019-06-07 16:04:08 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>Smith, John</td>
<td>qaok</td>
<td><a href="mailto:john.smith@onapp.com">john.smith@onapp.com</a></td>
<td>edgeserver, xiaakbnlihsynfg, xswkxlbdydnryx</td>
<td>17:26:53 +0300</td>
<td>00:16:3e:96:f9:03</td>
<td>xxswkxlbdydnryx</td>
</tr>
</tbody>
</table>

Where:

- **ip_address** - the IP address for which you want to generate the report
- **from** - start date during which an IP address was assigned to a user
- **to** - end date during which an IP address was assigned to a user
- **full_name** - user's first and last name
- **login** - the user to whom the IP address is assigned
- **email** - user email
- **hostname** - the name of your host
- **virtual_machine_identifier** - the VS identifier
- **network_interface_identifier** - the identifier of the network interface
- **mac_address** - MAC address of the network interface
45 IP Nets

This section contains the API requests you can use to manage the IP nets in your cloud. IP nets are a part of a network and contain ranges of IP addresses.

45.1 Get List of IP Nets in Network

To view the list of IP nets in a network, use the following request:

GET /settings/networks/:network_id/ip_nets.xml
GET /settings/networks/:network_id/ip_nets.json

XML Request Example

```
curl -i -X GET -u user:userpass -url
http://onapp.test/settings/networks/12/ip_nets.xml -H 'Accept:
application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X GET -u user:userpass -url
http://onapp.test/settings/networks/12/ip_nets.json -H 'Accept:
application/json' -H 'Content-type: application/json'
```

XML Output Example

```
<ip_nets type="array">
  <ip_net>
    <id type="integer">44</id>
    <network_address>192.168.72.0</network_address>
    <default_gateway>0.0.0.0</default_gateway>
    <network_mask type="integer">24</network_mask>
    <ipv4 type="boolean">true</ipv4>
    <label>72.x</label>
    <created_at type="dateTime">2019-08-08T00:03:55+03:00</created_at>
    <updated_at type="dateTime">2019-08-08T00:03:55+03:00</updated_at>
    <openstack_id nil="true"/>
    <kind>controlled</kind>
    <enabled type="boolean">true</enabled>
    <network>
      <id type="integer">36</id>
    </network>
  </ip_net>
</ip_nets>
```

Where:

- **id** - the ID of the IP net
- **network_address** - the network address of the IP net
- **default_gateway** - external gateway IP address
- **network_mask** - the network mask
- **ipv4** - whether this is an IPv4 or and IPv6 IP net: ’true’ for IPv4 networks and ’false’ for IPv6 networks
**45.2 Create IP Net**

To create an IP net in a network, use the following request:

**POST /settings/networks/:network_id/ip_nets.xml**

**POST /settings/networks/:network_id/ip_nets.json**

**XML Request Example**

```bash
curl -i -X POST -u user:userpass --url
'＜ip_net＞＜network_address＞192.168.0.0＜/network_address＞＜network_mask＞24＜/network_mask＞＜label＞AutoTestNetworkIpvNet1＜/label＞＜default_gateway＞2.2.2.1＜/default_gateway＞＜add_default_ip_range＞1＜/add_default_ip_range＞＜gateway_outside_ip_net＞1＜/gateway_outside_ip_net＞'＜/ip_net＞'
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass --url
http://onapp.test/settings/networks/12/ip_nets.json -H 'Accept: application/json' -H 'Content-type: application/json' -d "{'ip_net': {'network_address': '192.168.0.0', 'network_mask': '24', 'label': 'AutoTestNetworkIpvNet1', 'default_gateway': '2.2.2.1', 'add_default_ip_range': '1', 'gateway_outside_ip_net': '1'}}"
```

Where:

- **label** - the name of the IP net
- **network_mask** - the network mask
- **network_address** - the network address of the IP net
- **default_gateway** - the default gateway to be added to the IP net automatically
- **add_default_ip_range** - set this parameter to '1' for the default IP range to be added to the IP net automatically. Otherwise, set '0', then you'll need to add the required IP ranges after the IP net is created.
- **gateway_outside_ip_net** - set this parameter to '1' to allow the gateway to be outside from the IP net

**Page History**

v.6.1
- added the following parameters:
  - **default_gateway**
  - **gateway_outside_ip_net**
  - **add_default_ip_range**
45.3 Edit IP Net

To edit an IP net, use the following request:

PUT /settings/networks/:network_id/ip_nets/:ip_net_id.xml
PUT /settings/networks/:network_id/ip_nets/:ip_net_id.json

XML Request Example

```
curl -i -X PUT -u user:userpass --url
<ip_net><label>AutoTestNetworkIpvNet123</label><network_address>199.163.1.0</network_address><network_mask>24</network_mask></ip_net>'
```

JSON Request Example

```
curl -i -X PUT -u user:userpass --url
```

Where:

- **label** - the name of the IP net
- **network_mask** - the network mask. You cannot edit this parameter if the IP net contains IP ranges.
- **network_address** - the network address of the IP net. You cannot edit this parameter if the IP net contains IP ranges.

45.4 Delete IP Net

To delete an IP net, use the following request:

DELETE /settings/networks/:network_id/ip_nets/:ip_net_id.xml
DELETE /settings/networks/:network_id/ip_nets/:ip_net_id.json

XML Request Example

```
curl -i -X DELETE -u user:userpass --url
```

JSON Request Example

```
curl -i -X DELETE -u user:userpass --url
```
46 IP Ranges

This section contains the API requests you can use to manage the IP ranges in your cloud. IP ranges are a part of a network and contain IP addresses.

46.1 Get List of IP Ranges in IP Net

To view the list of the IP ranges in an IP net, use the following request:

GET /settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges.xml
GET /settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges.json

XML Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/settings/networks/12/ip_nets/44/ip_ranges.xml
'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/settings/networks/12/ip_nets/44/ip_ranges.json
'Accept: application/json' -H 'Content-type: application/json'
```

XML Output example

```
<ip_range type="array">
  <ip_range>
    <id type="integer">53</id>
    <start_address>192.168.72.1</start_address>
    <end_address>192.168.72.16</end_address>
    <default_gateway>192.168.72.1</default_gateway>
    <ipv4 type="boolean">true</ipv4>
    <created_at type="dateTime">2019-08-08T00:03:57+03:00</created_at>
    <updated_at type="dateTime">2019-08-08T00:03:57+03:00</updated_at>
  </ip_range>
</ip_range>
```

Where:

- **id** - the ID of the IP range
- **end_address** - the IP address with which your IP range ends
- **default_gateway** - the default gateway for the IP range
- **start_address** - the IP address with which your IP range starts
OnApp Cloud 6.3 API Guide

ipv4 - whether this is an IPv4 or and IPv6 IP range: `true` for IPv4 IP ranges and `false` for IPv6
IP ranges
created_at - the date in the [YYYY][MM][DD]T[hh][mm][ss] format
updated_at - the date in the [YYYY][MM][DD]T[hh][mm][ss] format
ip_net.id - IP net ID to which this IP ranges belong

46.2 Create IP Range
To create an IP range in a network, use the following request:
POST /settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges.xml
POST /settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges.json
XML Request Example
curl -i -X POST -u user:userpass --url
'Accept: application/xml' -H 'Content-type: application/xml' -d
'<ip_range><end_address>193.169.1.254</end_address><default_gateway>193.16
9.1.1</default_gateway><start_address>193.169.1.2</start_address><gateway_
outside_ip_net>1</gateway_outside_ip_net></ip_range>'

JSON Request Example
curl -i -X POST -u user:userpass --url
'Accept: application/json' -H 'Content-type: application/json' -d
'{"ip_range": {"end_address": "193.169.1.254", "default_gateway":
"193.169.1.1", "start_address": "193.169.1.2", "gateway_outside_ip_net":
"1"}}'

Where:
end_address - the IP address with which your IP range ends
default_gateway - the default gateway for the IP range
start_address - the IP address with which your IP range starts
gateway_ouside_ip_net - set this parameter to '1' to allow the gateway to be outside from the IP
net

Page History
v.6.1


added the gateway_ouside_ip_net parameter

46.3 Edit IP Range
To edit an IP range, use the following request:
PUT
/settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges/:ip_range_
id.xml

581


OnApp Cloud 6.3 API Guide

PUT
/settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges/:ip_range_
id.json
XML Request Example
curl -i -X PUT -u user:userpass --url
http://onapp.test/settings/networks/12/ip_nets/23/ip_ranges/1.xml -H
'Accept: application/xml' -H 'Content-type: application/xml' -d
'<ip_range><end_address>192.168.1.224</end_address><default_gateway>192.16
8.1.2</default_gateway><start_address>192.168.1.3</start_address></ip_rang
e>'

JSON Request Example
curl -i -X PUT -u user:userpass --url
http://onapp.test/settings/networks/12/ip_nets/23/ip_ranges/1.json -H
'Accept: application/json' -H 'Content-type: application/json' -d
'{"ip_range": {"end_address": "193.169.1.224", "default_gateway":
"193.169.1.2", "start_address": "193.169.1.3"}}'

Where:
end_address - the IP address with which your IP range ends
default_gateway - the default gateway for the IP range
start_address - the IP address with which your IP range starts

46.4 Delete IP Range
To delete an IP range, use the following request:
DELETE
/settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges/:ip_range_
id.xml
DELETE
/settings/networks/:network_id/ip_nets/:ip_net_id/ip_ranges/:ip_range_
id.json
XML Request Example
curl -i -X DELETE -u user:userpass --url
es/:ip_range_id.xml -H 'Accept: application/xml' -H 'Content-type:
application/xml'

JSON Request Example
curl -i -X DELETE -u user:userpass --url
es/:ip_range_id.json -H 'Accept: application/json' -H 'Content-type:
application/json'

582


47 ISOs

OnApp allows uploading your custom bootable ISOs for recovery purposes. These could be different images for Windows/Linux/FreeBSD or any additional software. You can boot virtual servers from your own ISOs or the ISOs that are uploaded and made publicly available by other users. Currently, there is a limitation of 1 GB for the ISOs uploaded from the file system. There are no space limitations for the ISOs uploaded from the URL (except for your disk space limitations).

- OnApp supports rebooting existing virtual and smart servers from ISO.
- To build a new server from an ISO, create a virtual server and then reboot this VS from the appropriate ISO.

47.1 Get List of ISOs

To view the list of ISOs, use the following request:

GET /template_isos.xml
GET /template_isos.json

XML Request Example


JSON Request Example


XML Output Example
Where:

label - the ISO title

min_memory_size - minimum memory size required to build a VS on this ISO (MB)

version - version of the file

operating_system - operating system name

operating_system_distro - operating system distribution

id - ID of the ISO

template_size - size of the ISO

virtualization - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO

allowed_hot_migrate - true if hot migration for VS, created from this ISO, is enabled; otherwise false

The following output parameters do not currently apply to ISOs:

allow_resize_without_reboot
allowed_swap
backup_server_id
baremetal_server
cdn
checksum
created_at
disk_target_device
ext4
file_name
initial_password
initial_username
min_disk_size
operating_system_arch
operating_system_edition
operating_system_tail
parent_template_id
remote_id
smart_server
state
updated_at
user_id

Page History
v.5.5
- added allowed_hot_migrate parameter

47.2 Get ISO Details

To view the details of the particular ISO, use the following request:

GET /template_isos/:iso_id.xml
GET /template_isos/:iso_id.json

XML Request Example


JSON Request Example

XML Output Example

```
<image_template_iso>
  <allow_resize_without_reboot nil="true"/>
  <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
  <allowed_swap type="boolean">true</allowed_swap>
  <backup_server_id nil="true"/>
  <baremetal_server type="boolean">false</baremetal_server>
  <cdn type="boolean">false</cdn>
  <checksum nil="true"/>
  <created_at type="datetime">2015-03-17T10:23:17+00:00</created_at>
  <ext4 type="boolean">false</ext4>
  <file_name>Fedora-20-x86_64-netinst.iso</file_name>
  <id type="integer">135</id>
  <initial_password nil="true"/>
  <initial_username nil="true"/>
  <label>Fedora-20-x86_64-netinst</label>
  <min_disk_size nil="true"/>
  <min_memory_size type="integer">512</min_memory_size>
  <operating_system>Linux</operating_system>
  <operating_system_arch nil="true"/>
  <operating_system_distro>redhat</operating_system_distro>
  <operating_system_edition nil="true"/>
  <operating_system_tail nil="true"/>
  <parent_template_id nil="true"/>
  <remote_id nil="true"/>
  <smart_server type="boolean">false</smart_server>
  <state>active</state>
  <template_size type="integer">1364</template_size>
  <updated_at type="datetime">2015-03-23T12:56+00:00</updated_at>
  <user_id nil="true"/>
  <version>1.0</version>
  <virtualization type="array">
    <virtualization>xen</virtualization>
    <virtualization>kvm</virtualization>
    <virtualization>kvm_virtio</virtualization>
  </virtualization>
</image_template_iso>
```

Where:

- **label** - the ISO title
- **min_memory_size** - minimum memory size required to build a VS on this ISO (MB)
- **version** - version of the file
- **operating_system** - operating system name
- **operating_system_distro** - operating system distribution
- **id** - ID of the ISO
- **template_size** - size of the ISO
- **virtualization** - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO
- **allowed_hot_migrate** - true if hot migration for VS, created from this ISO, is enabled; otherwise false

The following output parameters do not currently apply to ISOs:

- **allow_resize_without_reboot**
- **allowed_swap**
backup_server_id
baremetal_server
cdn
checksum
created_at
disk_target_device
ext4
file_name
initial_password
initial_username
min_disk_size
operating_system_arch
operating_system_edition
operating_system_tail
parent_template_id
remote_id
smart_server
state
updated_at
user_id

Page History
v.5.5
• added allowed_hot_migrate parameter

47.3 Get List of Public ISOs

To view the list of public ISOs, use the following request:

GET /template_isos/system.xml
GET /template_isos/system.json

XML Request Example

```
```

JSON Request Example

```
```
XML Output Example

```
<image_template_isos type="array">
  <image_template_iso>
    <allow_resize_without_reboot nil="true"/>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <baremetal_server type="boolean">false</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum nil="true"/>
    <created_at type="datetime">2015-03-17T10:23:17+00:00</created_at>
    <disk_target_device nil="true"/>
    <file_name>Fedora-20-x86_64-netinst.iso</file_name>
    <id type="integer">135</id>
    <initial_password nil="true"/>
    <initial_username nil="true"/>
    <label>Fedora-20-x86_64-netinst</label>
    <min_disk_size nil="true"/>
    <min_memory_size type="integer">512</min_memory_size>
    <operating_system>Linux</operating_system>
    <operating_system_arch nil="true"/>
    <operating_system_distro>redhat</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <smart_server type="boolean">false</smart_server>
    <state>active</state>
    <template_size type="integer">1364</template_size>
    <updated_at type="datetime">2015-03-23T12:12:56+00:00</updated_at>
    <user_id nil="true"/>
    <version>1.0</version>
  </image_template_iso>
  <virtualization type="array">
    <virtualization>xen</virtualization>
    <virtualization>kvm</virtualization>
    <virtualization>kvm_virtio</virtualization>
  </virtualization>
</image_template_iso>...
</image_template_isos type>
```

Where:

- **label** - the ISO title
- **min_memory_size** - minimum memory size required to build a VS on this ISO (MB)
- **version** - version of the file
- **operating_system** - operating system name
- **operating_system_distro** - operating system distribution
- **id** - ID of the ISO
- **template_size** - size of the ISO
- **virtualization** - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO
- **allowed_hot_migrate** - true if hot migration for VS, created from this ISO, is enabled; otherwise false

The following output parameters do not currently apply to ISOs:
allow_resize_without_reboot
allowed_swap
backup_server_id
baremetal_server
cdn
checksum
created_at
disk_target_device
ext4
file_name
initial_password
initial_username
min_disk_size
operating_system_arch
operating_system_edition
operating_system_tail
parent_template_id
remote_id
smart_server
state
updated_at
user_id

Page History
v.5.5
- added allowed_hot_migrate parameter

47.4 Get List of ISOs of Particular User

To view the list of ISOs that a particular user uploaded, use the following request:
GET /template_isos/user/:user_id.xml
GET /template_isos/user/:user_id.json

XML Request Example


JSON Request Example

XML Output Example

```
<image_template_isos type="array">
  <image_template_iso>
    <allow_resize_without_reboot nil="true"/>
    <allow_hot_migrate type="boolean">false</allow_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <baremetal_server type="boolean">false</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum nil="true"/>
    <created_at type="datetime">2015-03-17T10:23:17+00:00</created_at>
    <disk_target_device nil="true"/>
    <ext4 type="boolean">false</ext4>
    <file_name>Fedora-20-x86_64-netinst.iso</file_name>
    <id type="integer">135</id>
    <initial_password nil="true"/>
    <initial_username nil="true"/>
    <label>Fedora-20-x86_64-netinst</label>
    <min_disk_size nil="true"/>
    <min_memory_size type="integer">512</min_memory_size>
    <operating_system>Linux</operating_system>
    <operating_system_arch nil="true"/>
    <operating_system_distro>redhat</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <smart_server type="boolean">false</smart_server>
    <state>active</state>
    <template_size type="integer">1364</template_size>
    <updated_at type="datetime">2015-03-23T12:56+00:00</updated_at>
    <user_id>12</user_id>
    <version>1.0</version>
    <virtualization type="array">
      <virtualization>xen</virtualization>
      <virtualization>kvm</virtualization>
      <virtualization>kvm_virtio</virtualization>
    </virtualization>
  </image_template_iso>
</image_template_isos type>
```

Where:

- **label** - the ISO title
- **min_memory_size** - minimum memory size required to build a VS on this ISO (MB)
- **version** - version of the file
- **operating_system** - operating system name
- **operating_system_distro** - operating system distribution
- **id** - ID of the ISO
- **template_size** - size of the ISO
- **virtualization** - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO
allowed_hot_migrate - true if hot migration for VS, created from this ISO, is enabled; otherwise false

The following output parameters do not currently apply to ISOs:
allow_resize_without_reboot
allowed_swap
backup_server_id
baremetal_server
cdn
checksum
created_at
disk_target_device
ext4
file_name
initial_password
initial_username
min_disk_size
operating_system_arch
operating_system_edition
operating_system_tail
parent_template_id
remote_id
smart_server
state
updated_at
user_id

Page History
v.5.5
  • added allowed_hot_migrate parameter

47.5 Get List of User ISOs

To view the list of user ISOs, use the following request:
GET /template_isos/user.xml
GET /template_isos/user.json

XML Request Example

JSON Request Example


XML Output Example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<image_template_isos type="array">
  <image_template_iso>
    <allow_resize_without_reboot nil="true"/>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <baremetal_server type="boolean">false</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum nil="true"/>
    <created_at type="datetime">2015-03-17T10:23:17+00:00</created_at>
    <disk_target_device nil="true"/>
    <ext4 type="boolean">false</ext4>
    <file_name>Fedora-20-x86_64-netinst.iso</file_name>
    <id type="integer">135</id>
    <initial_password nil="true"/>
    <initial_username nil="true"/>
    <label>Fedora-20-x86_64-netinst</label>
    <min_disk_size nil="true"/>
    <min_memory_size type="integer">512</min_memory_size>
    <operating_system>Linux</operating_system>
    <operating_system_arch nil="true"/>
    <operating_system_distro>redhat</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <smart_server type="boolean">false</smart_server>
    <state>active</state>
    <template_size type="integer">1364</template_size>
    <updated_at type="datetime">2015-03-23T12:12:56+00:00</updated_at>
    <user_id nil="true"/>
    <version>1.0</version>
    <virtualization type="array">
      <virtualization>xen</virtualization>
      <virtualization>kvm</virtualization>
      <virtualization>kvm_virtio</virtualization>
    </virtualization>
  </image_template_iso>
</image_template_isos type>
```

Where:

- **label** - the ISO title
- **min_memory_size** - minimum memory size required to build a VS on this ISO (MB)
- **version** - version of the file
- **operating_system** - operating system name
**operating_system_distro** - operating system distribution

**id** - ID of the ISO

**template_size** - size of the ISO

**virtualization** - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO

**allowed_hot_migrate** - true if hot migration for VS, created from this ISO, is enabled; otherwise false

The following output parameters do not currently apply to ISOs:

- **allow_resize_without_reboot**
- **allowed_swap**
- **backup_server_id**
- **baremetal_server**
- **cdn**
- **checksum**
- **created_at**
- **disk_target_device**
- **ext4**
- **file_name**
- **initial_password**
- **initial_username**
- **min_disk_size**
- **operating_system_arch**
- **operating_system_edition**
- **operating_system_tail**
- **parent_template_id**
- **remote_id**
- **smart_server**
- **state**
- **updated_at**
- **user_id**

**Page History**

v.5.5

- added **allowed_hot_migrate** parameter

**47.6 Get List of Own ISOs**

To view the list of own ISOs, use the following request:

GET /template_isos/own.xml

GET /template_isos/own.json
### XML Request Example

```bash
```

### JSON Request Example

```bash
```

### XML Output Example

```xml
<image_template_isos type="array">
  <image_template_iso>
    <allow_resize_without_reboot nil="true"/>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <cdn type="boolean">false</cdn>
    <checksum nil="true"/>
    <created_at type="datetime">2015-03-17T10:23:17+00:00</created_at>
    <disk_target_device nil="true"/>
    <ext4 type="boolean">false</ext4>
    <file_name>Fedora-20-x86_64-netinst.iso</file_name>
    <id type="integer">135</id>
    <initial_password nil="true"/>
    <initial_username nil="true"/>
    <label>Fedora-20-x86_64-netinst</label>
    <min_disk_size nil="true"/>
    <min_memory_size type="integer">512</min_memory_size>
    <operating_system>Linux</operating_system>
    <operating_system_arch nil="true"/>
    <operating_system_distro>redhat</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <remote_id nil="true"/>
    <smart_server type="boolean">false</smart_server>
    <state>active</state>
    <template_size type="integer">1364</template_size>
    <updated_at type="datetime">2015-03-23T12:12:56+00:00</updated_at>
    <user_id nil="true"/>
    <version>1.0</version>
    <virtualization type="array">
      <virtualization>xen</virtualization>
      <virtualization>kvm</virtualization>
      <virtualization>kvm_virtio</virtualization>
    </virtualization>
  </image_template_iso>
  ...
</image_template_isos type>
```

Where:
- **label** - the ISO title
- **min_memory_size** - minimum memory size required to build a VS on this ISO (MB)
- **version** - version of the file
- **operating_system** - operating system name
operating_system_distro - operating system distribution
id - ID of the ISO
template_size - size of the ISO
virtualization - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO
allowed_hot_migrate - true if hot migration for VS, created from this ISO, is enabled; otherwise false

The following output parameters do not currently apply to ISOs:
allow_resize_without_reboot
allowed_swap
backup_server_id
baremetal_server
cdn
checksum
created_at
disk_target_device
ext4
file_name
initial_password
initial_username
min_disk_size
operating_system_arch
operating_system_edition
operating_system_tail
parent_template_id
remote_id
smart_server
state
updated_at
user_id

Page History
v.5.5
• added allowed_hot_migrate parameter

47.7 Update ISO

To update an ISO, use the following request:
PUT /template_isos/:id.xml
PUT /template_isos/:id.json
**XML Request Example**

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<image_template_iso><label>label</label><min_disk_size>10</min_disk_size>
  <min_memory_size>320</min_memory_size><version>1.3</version><operating_system>Linux</operating_system>
  <operating_system_distro>redhat</operating_system_distro><virtualization>xen</virtualization>
  <allowed_hot_migrate>true</allowed_hot_migrate></image_template_iso>' --url
  http://onapp.test/template_isos/2.xml
```

**JSON Request Example**

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"image_template_iso":{"label":"label","version":"1.3","min_memory_size":
  "320","min_disk_size": "10","operating_system": "Linux",
  "operating_system_distro": "Fedore","allowed_hot_migrate": true,
  "virtualization": ["xen", "kvm"]}}' --url
  http://onapp.test/template_isos/2.json
```

Where:

- **label** – the ISO title
- **min_memory_size** – minimum memory size required to build a VS on this ISO (MB)
- **version** – file version
- **operating_system** - operating system name
- **operating_system_distro** - type in the operating system distribution in free form
- **min_disk_size** - the minimum disk size required to build a VS on this ISO (GB)
- **virtualization** - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO
- **allowed_hot_migrate** - set true to enable hot migration for VS created from this ISO

**Page History**

v.5.5

- added **allowed_hot_migrate** parameter

**47.8 Add New ISO**

To add a new ISO, use the following request:

- POST /template_isos.xml
- POST /template_isos.json

**XML Request Example**

```xml
<image_template_iso>
  <label>label</label>
  <min_disk_size>10</min_disk_size>
  <min_memory_size>320</min_memory_size>
  <version>1.3</version>
  <operating_system>Linux</operating_system>
  <operating_system_distro>redhat</operating_system_distro>
  <virtualization>xen</virtualization>
  <allowed_hot_migrate>true</allowed_hot_migrate>
</image_template_iso>
```

**JSON Request Example**


**Where:**

- **make_public** - whether the ISO will be available to all users
- **label** - the ISO title
- **min_memory_size** - minimum memory size required to build a VS on this ISO (MB)
- **version** - version of the file
- **operating_system** - operating system name
- **operating_system_distro** - type in the operating system distribution in free form
- **virtualization** - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this ISO
- **min_disk_size** - the minimum disk size required to build a VS on this ISO (GB)
- **file_url** - the URL from which a file with the .iso extension is to be uploaded
- **allowed_hot_migrate** - set true to enable hot migration for VS created from this ISO

**Page History**

v.5.5

- added **allowed_hot_migrate** parameter

**47.9 Make ISO Public**

To make an ISO public, use the following request:

POST /template_isos/:id/make_public.xml
POST /template_isos/:id/make_public.json

**XML Request Example**
JSON Request Example


If an ISO is queued to be moved to a public list successfully, an HTTP 201 response is returned.

Only User ISOs can be made public.

47.10 Delete ISO

To delete an ISO from the system, use the following request:

DELETE /template_isos/:id.xml
DELETE /template_isos/:id.json

XML Request Example


JSON Request Example


The system won't delete the ISO if it is used by any VSs.
48 License

How to view and edit the details on the OnApp license.

48.1 Get License Details

To see the license details, use the following request:

GET /settings/license.xml
GET /settings/license.json

XML Request Example


JSON Request Example


XML Output Example

```xml
<license>
  <type>TEST</type>
  <key>XXXXX-XXXXX-XXXXX-XXXXX-XXXXX</key>
  <valid>true</valid>
  <status>Valid</status>
  <kvm_xen_hv_limit>14</kvm_xen_hv_limit>
  <kvm_xen_vm_limit>9999</kvm_xen_vm_limit>
  <vcenter_vm_limit>9999</vcenter_vm_limit>
  <kvm_xen_core_limit>9999</kvm_xen_core_limit>
  <vcenter_core_limit>9999</vcenter_core_limit>
  <integrated_storage_limit>9999</integrated_storage_limit>
  <trader_status>Enabled</trader_status>
  <trader_allowed>false</trader_allowed>
  <supplier_status>Enabled</supplier_status>
  <supplier_allowed>false</supplier_allowed>
</license>
```

Where:

- **type** – the type of the license
- **key** – the key of the license
- **valid** – true, if the license is valid, otherwise, false
- **status** – the status of the license that can be **valid** or **invalid**
- **kvm_xen_hv_limit** – the limit on XEN/KVM compute resources
- **kvm_xen_vm_limit** – the limit on the number of virtual servers on XEN/KVM compute resources
- **vcenter_vm_limit** – the limit on the number of virtual servers on vCenter compute resources
- **kvm_xen_core_limit** – the core limit on XEN/KVM compute resources
vcenter_core_limit – the core limit on vCenter compute resources
integrated_storage_limit – the limit on the integrated storage disk size measured in GB
trader_status – the status of the trader
trader_allowed – true, if the trader is allowed to use this license (applicable for Federation)
supplier_status – the status of the supplier
supplier_allowed – true, if the supplier is allowed to use this license (applicable for Federation)

Page History
v. 5.6
• added the following parameters:
  o kvm_xen_hv_limit
  o kvm_xen_vm_limit
  o vcenter_vm_limit
  o kvm_xen_core_limit
  o vcenter_core_limit
  o integrated_storage_limit
• removed the core_limit parameter

48.2 Edit License Details
To update a license, use the following request:
PUT /settings.xml
PUT /settings.json
XML Request Example
```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '
<configuration><isolated_license>false</isolated_license><license_key>NNNN-NNNN-NNNN-NNNN-NNNN-NNNN</license_key></configuration>' --url http://onapp.test/settings.xml
```

JSON Request Example
```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '
{"configuration":{"isolated_license":"true","license_key":"NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN-NNNN"}}' --url http://onapp.test/settings.json
```

Where:
isolated_license – true, if the isolated license is used on the CP, otherwise, false
license_key – the key of your OnApp license

Page History
v. 5.6
• added the isolated_license parameter
49 Load Balancers

Load Balancers, both autoscaling clusters and load balancer clusters, can only be created on the basis of Virtual Servers, and are not available for Smart Servers, Baremetal Servers, or VMware Virtual Servers.

Load balancers distribute requests evenly between clustered virtual servers (nodes), so that no virtual server is overloaded. Together with nodes, load balancers form Load Balancing Clusters. There are two options of load balancing clusters:

Cluster type
In this case you specify which VSs (nodes) will participate in a load balancing cluster. You can add and remove clustered VSs as required.

Autoscaling type
In this case you indicate minimum and maximum number of nodes for a cluster, as well as autoscaling parameters for automatic adding or removing nodes from the cluster. The system creates required number of identical nodes, with the same resource allocation and the same template for each node.

Load balancing clusters of both types use the same requests. Only some parameters differ.

49.1 Get List of Load Balancers

To get the list of available load balancers, use the following request:

GET /load_balancers.xml
GET /load_balancers.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example
<load_balancers type="array">
  <load_balancer>
    <admin_note nil="true"/>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <admin_note nil="true"/>
    <allow_resize_without_reboot type="boolean">true</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <booted type="boolean">true</booted>
    <built type="boolean">true</built>
    <cpu_shares type="integer">10</cpu_shares>
    <cpus type="integer">1</cpus>
    <created_at type="datetime">2013-08-01T18:37:03:00</created_at>
    <customer_network_id nil="true"/>
    <deleted_at nil="true"/>
    <edge_server_type nil="true"/>
    <enable_monitis type="boolean">false</enable_monitis>
    <firewall_notrack type="boolean">false</firewall_notrack>
    <hostname>zaza</hostname>
    <hypervisor_id type="integer">3</hypervisor_id>
    <id type="integer">1654</id>
    <identifier>pop7ba0j4imc7e</identifier>
    <initial_root_password>Mvhn1gUjXpdS</initial_root_password>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <label>zaza_B</label>
    <local_remote_access_ip_address>109.123.91.38</local_remote_access_ip_address>
    <local_remote_access_port type="integer">5904</local_remote_access_port>
    <locked type="boolean">false</locked>
    <memory type="integer">512</memory>
    <min_disk_size type="integer">5</min_disk_size>
    <operating_system>linux</operating_system>
    <operating_system_distro>lbva</operating_system_distro>
    <preferred_hvs type="array"/>
    <recovery_mode nil="true"/>
    <remote_access_password>Y7wuNG1EpkZO</remote_access_password>
    <service_password nil="true"/>
    <state>new</state>
    <storage_server_type nil="true"/>
    <strict_virtual_machine_id nil="true"/>
    <suspended type="boolean">false</suspended>
    <template_id type="integer">10</template_id>
    <template_label>Load Balancer Virtual Appliance</template_label>
    <updated_at type="datetime">2013-08-01T18:43:01+03:00</updated_at>
    <user_id type="integer">337</user_id>
    <vip nil="true"/>
    <xen_id type="integer">215</xen_id>
  </load_balancer>
</load_balancers>
Where:

load balancer - an array of load balancer details
add_to_marketplace – this parameter is not applicable to load balancers
admin_note – an optional text note
allow_resize_without_reboot – true if you can resize a VS’s CPU and RAM without rebooting it
allowed_hot_migrate – true if hot migration is allowed
allowed_swap – true if swap disks are allowed, otherwise false
booted - true if the server is booted, otherwise false
built – true if the load balancing cluster is built, otherwise false
cpu_shares – the CPU priority of this load balancing cluster
cpus – the number of CPU cores allocated to this load balancer
created_at – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
customer_network_id - this parameter is not applicable to load balancers

deleted_at the date in the [YYYY][MM][DD][hh][mm][ss]Z format
edge_server_type - this parameter is not applicable to load balancers
enable_autoscale – true if autoscaling is enabled, otherwise false
firewall_notrack - true if the NOTRACK rule is set in iptables
hostname – the load balancer host name
hypervisor_id – IDs of the compute resources used by this load balancing cluster
id – the load balancing cluster ID
identifier – identifier of the load balancer in the database
initial_root_password — the LB root password
initial_root_password_encrypted - true, if the root password is encrypted, otherwise false
label – the load balancer name
local_remote_access_ip_address - IP address used for remote access
local_remote_access_port – the port ID used for console access
locked – true if locked, otherwise false
memory – the amount of RAM allocated to this load balancing cluster
min_disk_size – the minimum disk size in GB required for a specified template
note – an optional text, added as a note
operating_system - the OS on which the load balancing cluster is based
operating_system_distro – the distribution of the OS
preferred_hvs - the array of preferable compute resources based on compute zone that meet some load balancer configuration settings
recovery_mode – true if recovery mode is allowed, otherwise false
remote_access_password – the password for remote access
service_password – this parameter is not applicable to load balancers
state – deprecated attribute
storage_server_type – this parameter is not applicable to load balancers
strict_virtual_machine_id – the ID of a VS that will never reside in this load balancing cluster
suspended – true if suspended, otherwise false
template_id – ID of the LB template
template_label - the name of the template on which this load balancing cluster is based
updated_at - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
user_id – the ID of the user who owns this load balancing cluster

vip – true if the VIP status is set, otherwise false
xen_id - the VS ID set by the virtualization engine

ip_addresses - an array of IP addresses assigned to this load balancer and their details:
- address – IP address
- broadcast – broadcast address
- created_at — the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- customer_network_id - the ID of the customer VLAN the IP address belongs to
- disallowed_primary – true if not allowed to be used as primary, otherwise false
- gateway – gateway address
- hypervisor_id - the ID of a compute resource the IP address is associated with
- id –the ID of the IP address
- `ip_address_pool_id` - ID of the IP address pool the IP address is associated with
- `network_address` – the address of the network
- `network_id` – the ID of the network
- `pxe` - true, if this address can be used for cloudbooting a compute resource
- `monthly_bandwidth_used` - LB monthly bandwidth
- `total_disk_size` - total LB disk size
- `cpu_priority` - the CPU priority for the LB; it has the same value as `cpu_shares` parameter
- `price_per_hour` - price per hour set for this load balancer
- `price_per_hour_powered_off` - price per hour set for this load balancer in the powered off state

**Page History**

v.6.1
- added the `cpu_priority` parameter

### 49.2 Get Load Balancer Details

To get the load balancer details, use the following request:

GET `/load_balancers/:load_balancer_id.xml`
GET `/load_balancers/:load_balancer_id.json`

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
<load_balancer>
  <id type="integer">1741</id>
  <hypervisor_id type="integer">12</hypervisor_id>
  <template_id type="integer">1</template_id>
  <identifier>nkmecicwkgktvx</identifier>
  <hostname>fgdfbf</hostname>
  <memory type="integer">512</memory>
  <cpus type="integer">1</cpus>
  <cpu_shares type="integer">1</cpu_shares>
  <created_at type="dateTime">2019-02-20T14:19:30+02:00</created_at>
  <updated_at type="dateTime">2019-02-20T14:29:51+02:00</updated_at>
  <built type="boolean">true</built>
  <locked type="boolean">false</locked>
  <booted type="boolean">false</booted>
  <remote_access_password>9eGG6YkoM6Yx</remote_access_password>
  <local_remote_access_port nil="true"/>
  <label>autoscale</label>
  <recovery_mode type="boolean">false</recovery_mode>
  <user_id type="integer">1088</user_id>
  <operating_system>linux</operating_system>
  <operating_system_distro>lbva</operating_system_distro>
  <allowed_swap type="boolean">true</allowed_swap>
  <template_label>Load Balancer Virtual Appliance</template_label>
  <min_disk_size type="integer">5</min_disk_size>
  <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
  <note nil="true"/>
  <admin_note nil="true"/>
  <suspended type="boolean">false</suspended>
  <strict_virtual_machine_id nil="true"/>
  <enable_autoscale type="boolean">false</enable_autoscale>
  <add_to_marketplace nil="true"/>
  <state>failed</state>
  <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
  <storage_server_type nil="true"/>
  <firewall_notrack type="boolean">false</firewall_notrack>
  <service_password nil="true"/>
  <preferred_hvs type="array"/>
  <local_remote_access_ip_address>109.123.125.5</local_remote_access_ip_address>
  <cpu_units type="integer">10</cpu_units>
  <cpu_sockets nil="true"/>
  <draas_keys type="array"/>
  <iso_id nil="true"/>
  <cores_per_socket type="integer">1</cores_per_socket>
  <instance_package_id nil="true"/>
  <hot_add_cpu nil="true"/>
  <hot_add_memory nil="true"/>
  <time_zone nil="true"/>
  <autoscale_service nil="true"/>
  <cdboot type="boolean">false</cdboot>
  <draas_mode type="integer">0</draas_mode>
  <vapp_id nil="true"/>
  <vmware_tools nil="true"/>
  <vcenter_moref nil="true"/>
  <template_version>1.7</template_version>
  <openstack_id nil="true"/>
  <domain>localdomain</domain>
  <vcenter_reserved_memory type="integer">0</vcenter_reserved_memory>
  <deleted_at nil="true"/>
  <properties> </properties>
  <acceleration_allowed type="boolean">true</acceleration_allowed>
  <vcenter_cluster_id nil="true"/>
</load_balancer>
<ip_addresses type="array">
  <networking_ip_address>
    <id type="integer">1756</id>
    <address>109.123.91.151</address>
    <broadcast>109.123.91.191</broadcast>
    <network_address>109.123.91.128</network_address>
    <gateway>109.123.91.129</gateway>
    <created_at type="dateTime">2019-02-20T14:19:30+02:00</created_at>
    <updated_at type="dateTime">2019-02-20T14:19:30+02:00</updated_at>
    <user_id nil="true"/>
    <pxe type="boolean">false</pxe>
    <hypervisor_id nil="true"/>
    <ip_range_id type="integer">1</ip_range_id>
    <external_address nil="true"/>
    <free type="boolean">false</free>
    <netmask>255.255.255.192</netmask>
  </networking_ip_address>

  <networking_ip_address>
    <id type="integer">1757</id>
    <address>109.123.91.152</address>
    <broadcast>109.123.91.191</broadcast>
    <network_address>109.123.91.128</network_address>
    <gateway>109.123.91.129</gateway>
    <created_at type="dateTime">2019-02-20T14:19:30+02:00</created_at>
    <updated_at type="dateTime">2019-02-20T14:19:30+02:00</updated_at>
    <user_id nil="true"/>
    <pxe type="boolean">false</pxe>
    <hypervisor_id nil="true"/>
    <ip_range_id type="integer">1</ip_range_id>
    <external_address nil="true"/>
    <free type="boolean">false</free>
    <netmask>255.255.255.192</netmask>
  </networking_ip_address>

  <networking_ip_address>
    <id type="integer">1758</id>
    <address>109.123.105.178</address>
    <broadcast>109.123.105.191</broadcast>
    <network_address>109.123.105.176</network_address>
    <gateway>109.123.105.177</gateway>
    <created_at type="dateTime">2019-02-20T14:19:30+02:00</created_at>
    <updated_at type="dateTime">2019-02-20T14:19:30+02:00</updated_at>
    <user_id nil="true"/>
    <pxe type="boolean">false</pxe>
    <hypervisor_id nil="true"/>
    <ip_range_id type="integer">4</ip_range_id>
    <external_address nil="true"/>
    <free type="boolean">false</free>
    <netmask>255.255.255.240</netmask>
  </networking_ip_address>
</ip_addresses>

<monthly_bandwidth_used type="decimal">0.0</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>
<support_incremental_backups type="boolean">false</support_incremental_backups>
<cpu_priority type="integer">1</cpu_priority>
<built_from_iso type="boolean">false</built_from_iso>
<acceleration type="boolean">false</acceleration>
<acceleration_status>Inactive</acceleration_status>
<hypervisor_type>kvm</hypervisor_type>
</load_balancer>

Where:

id - the ID of the load balancer
built - true if the load balancing cluster is built, otherwise false
remote_access_password - the password for remote access
suspended - true if suspended, otherwise false
strict_virtual_machine_id - the ID of a VS that will never reside in this load balancing cluster
enable_autoscale - true if autoscaling is enabled, otherwise false
add_to_marketplace - this parameter is not applicable to load balancers
state - deprecated attribute
initial_root_password_encrypted - true, if the root password is encrypted, otherwise false
edge_server_type - this parameter is not applicable to load balancers
storage_server_type - this parameter is not applicable to load balancers
firewall_notrack - true if the NOTRACK rule is set in iptables
service_password - this parameter is not applicable to load balancers
preferred_hvs - the array of preferable compute resources based on compute zone that meet some load balancer configuration settings
local_remote_access_ip_address - IP address used for remote access
cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in user billing plan
cpu_sockets - the amount of CPU sockets per core
ip_addresses - an array of IP addresses assigned to this load balancer and their details:
  • networking_ip_address - the ID of the network
  • id - the ID of the IP address
  • address – IP address
  • broadcast – broadcast address
  • network_address – the address of the network
  • gateway – gateway address
  • created_at — the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  • user_id - the ID of the user VLAN the IP address belongs to
  • pxe - true, if this address can be used for cloudbooting a compute resource
  • hypervisor_id - the ID of a compute resource the IP address is associated with
  • free - true if free, otherwise false
  • netmask - netmask for the IP address
monthly_bandwidth_used - LB monthly bandwidth
total_disk_size - the total disk size in GB of all disks assigned to the LB
support_incremental_backups - true if the LB supports incremental backups, and false if it does not
cpu_priority - it has the same value as cpu_shares parameter
hypervisor_type - the type of the compute resource, can be Xen or KVM

49.3 Get the List of Load Balancing Clusters

To get the list of load balancing clusters, use the following request:
GET /load_balancing_clusters.xml
GET /load_balancing_clusters.json

Load balancing cluster array includes details on load balancers and attached nodes.

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**
<load_balancing_clusters type="array">
  <load_balancing_cluster>
    <cluster_type>autoscaleout</cluster_type>
    <config>
      <max_node_amount type="integer">4</max_node_amount>
      <min_node_amount type="integer">2</min_node_amount>
    </config>
    <created_at type="datetime">2013-08-05T10:58:42+03:00</created_at>
    <id type="integer">26</id>
    <identifier>b5886a2f0c7b811992b72b82cadddee501c7f49c31</identifier>
    <image_template_id type="integer">1</image_template_id>
    <load_balancer_id type="integer">1669</load_balancer_id>
    <load_balancer_password>gPo96LEBwjWI</load_balancer_password>
    <name>az_AS</name>
    <node_attributes>
      <cpus>2</cpus>
      <cpu_shares>2</cpu_shares>
      <memory>256</memory>
      <rate_limit>50</rate_limit>
    </node_attributes>
    <created_at type="datetime">2013-08-05T10:58:44+03:00</created_at>
    <id type="integer">31</id>
    <ip_address_id type="integer">10</ip_address_id>
  </load_balancing_cluster_node>
  <nodes type="array">
    <load_balancing_cluster_node>
      <cluster_id type="integer">26</cluster_id>
      <created_at type="datetime">2013-08-05T10:58:44+03:00</created_at>
      <id type="integer">31</id>
      <ip_address_id type="integer">10</ip_address_id>
      <created_at type="datetime">2013-08-05T12:27:21+03:00</created_at>
      <id type="integer">337</id>
    </load_balancing_cluster_node>
    <ports type="array">
      <port type="integer">80</port>
      <port type="integer">345</port>
      <port type="integer">678</port>
    </ports>
  </nodes>
  <load_balancer>
    <add_to_marketplace nil="true"/>
    <admin_note nil="true"/>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <booted type="boolean">true</booted>
    <built type="boolean">true</built>
    <cpu_shares type="integer">10</cpu_shares>
    <cpus type="integer">1</cpus>
    <created_at type="datetime">2013-08-01T18:13:37+03:00</created_at>
    <customer_network_id nil="true"/>
    <deleted_at nil="true"/>
    <edge_server_type nil="true"/>
    <enableautoscale nil="true"/>
    <enable_monitis type="boolean">false</enable_monitis>
    <firewall_notrack type="boolean">false</firewall_notrack>
    <hostname>zaza</hostname>
    <hypervisor_id type="integer">3</hypervisor_id>
    <id type="integer">1654</id>
    <identifier>pop7ba0j4imc7e</identifier>
    <initial_root_password>Mvhn1gUjXpdS</initial_root_password>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <label>zaza_B</label>
    <local_remote_access_ip_address>109.123.91.38</local_remote_access_ip_address>
  </load_balancer>
</load_balancing_clusters>
<local_remote_access_port type="integer">5904</local_remote_access_port>
<locked type="boolean">false</locked>
<min_disk_size type="integer">5</min_disk_size>
<note nil="true"/>
<operating_system>linux</operating_system>
<operating_system_distro>lbva</operating_system_distro>
<preferred_hvs type="array"/>
<recovery_mode nil="true"/>
<remote_access_password>Y7wuNG1EpkZO</remote_access_password>
<service_password nil="true"/>
<state><new/></state>
<storage_server_type nil="true"/>
<strict_virtual_machine_id nil="true"/>
<suspended type="boolean">false</suspended>
<templat_id type="integer">10</template_id>
<operating_system>linux</operating_system>
<operating_system_distro>lbva</operating_system_distro>
<preferred_hvs type="array"/>
<recovery_mode nil="true"/>
<remote_access_password>Y7wuNG1EpkZO</remote_access_password>
<service_password nil="true"/>
<state><new/></state>
<storage_server_type nil="true"/>
<strict_virtual_machine_id nil="true"/>
<suspended type="boolean">false</suspended>
<templat_id type="integer">10</template_id>
<operating_system>linux</operating_system>
<operating_system_distro>lbva</operating_system_distro>
<preferred_hvs type="array"/>
<recovery_mode nil="true"/>
<remote_access_password>Y7wuNG1EpkZO</remote_access_password>
<service_password nil="true"/>
```xml
<created_at type="datetime">2013-08-05T10:58:42+03:00</created_at>
<enabled type="boolean">true</enabled>
<for_minutes type="integer">10</for_minutes>
<id type="integer">58</id>
<units type="integer">2</units>
<updated_at type="datetime">2013-08-05T11:42:25+03:00</updated_at>
</auto_scaling_out_memory>

<auto_scaling_out_cpu>
<created_at type="datetime">2013-08-05T10:58:42+03:00</created_at>
<enabled type="boolean">true</enabled>
<for_minutes type="integer">10</for_minutes>
<id type="integer">57</id>
<units type="integer">2</units>
<updated_at type="datetime">2013-08-05T11:42:25+03:00</updated_at>
</auto_scaling_out_cpu>

<auto_scaling_in_cpu>
<created_at type="datetime">2013-08-05T10:58:42+03:00</created_at>
<enabled type="boolean">true</enabled>
<for_minutes type="integer">30</for_minutes>
<id type="integer">59</id>
<units type="integer">2</units>
<updated_at type="datetime">2013-08-05T11:42:25+03:00</updated_at>
</auto_scaling_in_cpu>

<auto_scaling_in_memory>
<created_at type="datetime">2013-08-05T10:58:42+03:00</created_at>
<enabled type="boolean">true</enabled>
<for_minutes type="integer">30</for_minutes>
<id type="integer">60</id>
<units type="integer">2</units>
<updated_at type="datetime">2013-08-05T11:42:25+03:00</updated_at>
</auto_scaling_in_memory>
</load_balancing_cluster>
</load_balancing_clusters>

Where:

- **load_balancing_clusters** – an array of all load balancing clusters (may be both cluster and autoscaling types)
- **cluster_type** – the type of the cluster (either cluster or autoscaleout)
- **config** – configuration array, where:
  - **max_node_amount** – maximum number of nodes (for autoscaling types; remains empty for cluster types)
  - **min_node_amount** – minimum number of nodes (for autoscaling types; remains empty for cluster types)
- **created_at** – the date when the cluster was created
- **id** – ID of the cluster
- **identifier** – the LB identifier in the DB
- **image_template_id** – the ID of a template on which the nodes of this load balancer are based (empty for cluster type)
- **load_balancer_id** - the ID of a load balancer added to this cluster
- **load_balancer_password** – root password, which is generated automatically
- **name** - load balancing cluster name
node_attributes – an array of node attributes for autoscaling type, including cpu_shares, memory (RAM), rate_limit (port speed) and cpus (remains empty for cluster type)

- cpus – the number of CPU cores allocated to this load balancer
- cpu_shares – the CPU priority of this load balancing cluster
- memory – the amount of RAM allocated to this load balancing cluster
- rate_limit - the port speed, set for the LB

updated_at – the date when the cluster was updated

user_id – ID of the load balancing cluster owner

nodes - an array of load balancing cluster nodes with VS details:

- created_at – the date when the cluster node was created
- cluster_id - the ID of load balancing cluster to which this node belongs
- ip_address_id – the ID of VS IP address added to a cluster
- id – node ID
- updated_at – the date when the cluster node was updated
- virtual_machine_id – the ID of VS added to a cluster

ports – the array of ports on which this cluster runs

- port – the cluster port

load_balancer - an array of load balancer details:

- add_to_marketplace – this parameter is not applicable to load balancers
- admin_note – an optional text note
- allow_resize_without_reboot – true if you can resize a VS's CPU and RAM without rebooting it
- allowed_hot_migrate – true if hot migration is allowed
- allowed_swap – true if swap disks are allowed, otherwise false
- booted - true if the server is booted, otherwise false
- built – true if the load balancing cluster is built, otherwise false
- cpu_shares – the CPU priority of this load balancing cluster
- cpus – the number of CPU cores allocated to this load balancer
- created_at – the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- customer_network_id - this parameter is not applicable to load balancers
- deleted_at the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- edge_server_type -this parameter is not applicable to load balancers
- enable_autoscale – true if autoscaling is enabled, otherwise false
- firewall_notrack - true if the NOTRACK rule is set in iptables
- hostname – the load balancer host name
- hypervisor_id – IDs of the compute resources used by this load balancing cluster
- id – the load balancing cluster ID
- identifier – identifier of the load balancer in the database
- initial_root_password — the LB root password
- `initial_root_password_encrypted` - true, if the root password is encrypted, otherwise false
- `label` - the load balancer name
- `local_remote_access_ip_address` - IP address used for remote access
- `local_remote_access_port` - the port ID used for console access
- `locked` - true if locked, otherwise false
- `memory` - the amount of RAM allocated to this load balancing cluster
- `min_disk_size` - the minimum disk size in GB required for a specified template
- `note` - an optional text, added as a note
- `operating_system` - the OS on which the load balancing cluster is based
- `operating_system_distro` - the distribution of the OS
- `preferred_hvs` - the array of preferable compute resources based on compute zone that meet some load balancer configuration settings
- `recovery_mode` - true if recovery mode is allowed, otherwise false
- `remote_access_password` - the password for remote access
- `service_password` - this parameter is not applicable to load balancers
- `state` - deprecated attribute
- `storage_server_type` - this parameter is not applicable to load balancers
- `strict_virtual_machine_id` - the ID of a VS that will never reside in this load balancing cluster
- `suspended` - true if suspended, otherwise false
- `template_id` - ID of the LB template
- `template_label` - the name of the template on which this load balancing cluster is based
- `updated_at` - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- `user_id` - the ID of the user who owns this load balancing cluster
- `vip` - true if the VIP status is set, otherwise false
- `xen_id` - the VS ID set by the virtualization engine
- `ip_addresses` - an array of IP addresses assigned to this load balancer and their details:
  - `address` - IP address
  - `broadcast` - broadcast address
  - `created_at` - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
  - `customer_network_id` - the ID of the customer VLAN the IP address belongs to
  - `disallowed_primary` - true if not allowed to be used as primary, otherwise false
  - `gateway` - gateway address
  - `hypervisor_id` - the ID of a compute resource the IP address is associated with
  - `id` - the ID of the IP address
  - `ip_address_pool_id` - ID of the IP address pool the IP address is associated with
  - `network_address` - the address of the network
  - `network_id` - the ID of the network
  - `pxe` - true, if this address can be used for cloudbooting a compute resource
- `monthly_bandwidth_used` - LB monthly bandwidth
- `total_disk_size` - total LB disk size
- `price_per_hour` - price per hour set for this load balancer
- `price_per_hour_powered_off` - price per hour set for this load balancer in the powered off state
- `cpu_priority` - the CPU priority of this load balancing cluster; it has the same value as `cpu_shares` parameter

`auto_scaling_out_cpu` - an array of CPU autoscale out settings defining when the system should add more nodes to this autoscaling cluster:
- `created_at` - time when the CPU autoscale out settings were set
- `enabled` - true, if enabled, otherwise false
- `for_minutes` - the time threshold before scaling will be triggered
- `id` - parameter ID
- `units` - an amount of nodes that will be added when the `value` limit is reached
- `updated_at` - time when the CPU autoscale out settings were updated
- `value` - if the CPU usage is above percentage. An amount of nodes specified in `units` parameter will be added until the limit specified is reached.

`auto_scaling_in_cpu` - an array of CPU autoscale in settings:
- `created_at` - time when the CPU autoscale in settings were set
- `enabled` - true, if enabled, otherwise false
- `for_minutes` - the time threshold before scaling will be triggered
- `id` - parameter ID
- `units` - an amount of nodes that will be removed when the `value` limit is reached
- `updated_at` - time when the CPU autoscale in settings were updated
- `value` - if the CPU usage is less percentage. An amount of nodes specified in `units` parameter will be removed until the limit specified is reached.

`auto_scaling_in_memory` - an array of memory autoscale in settings:
- `created_at` - time when the memory autoscale in settings were set
- `enabled` - true, if enabled, otherwise false
- `for_minutes` - the time threshold before scaling will be triggered
- `id` - parameter ID
- `units` - an amount of nodes that will be removed when the `value` limit is reached
- `updated_at` - time when the memory autoscale in settings were updated
• *value* - if the memory usage is less percentage. An amount of nodes specified in *units* parameter will be removed until the limit specified is reached.

### 49.4 Get Load Balancing Cluster Details

To get details for a particular load balancing cluster, use the following request:

```
GET /load_balancing_clusters/:id.xml
GET /load_balancing_clusters/:id.json
```

Load balancing cluster array includes details on load balancers and attached nodes.

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```xml
<load_balancing_cluster>
  <name>Example Cluster</name>
  <description>Cluster for load balancing example</description>
  <nodes>
    <node id="1" address="192.168.1.1"/>
    <node id="2" address="192.168.1.2"/>
  </nodes>
  <load_balancers>
    <load_balancer id="1" type="roundrobin" />
  </load_balancers>
</load_balancing_cluster>
```
<load_balancing_cluster>
<cluster_type>autoscaleout</cluster_type>
<config>
<max_node_amount type="integer">4</max_node_amount>
<min_node_amount type="integer">2</min_node_amount>
</config>
<created_at type="datetime">2013-08-05T10:58:42+03:00</created_at>
<id type="integer">26</id>
<identifier>b5886a2f0c7b811992b72b82cadee501c7f49c31</identifier>
<image_template_id type="integer">1</image_template_id>
<load_balancer_id type="integer">1669</load_balancer_id>
<load_balancer_password>gPo96LEwjWI</load_balancer_password>
<name>az_AS</name>
<node_attributes>
<cpus>2</cpus>
<cpu_shares>2</cpu_shares>
<memory>256</memory>
<rate_limit>50</rate_limit>
</node_attributes>
<created_at type="datetime">2013-08-05T10:58:44+03:00</created_at>
<id type="integer">31</id>
<ip_address_id type="integer">10</ip_address_id>
<updated_at type="datetime">2013-08-05T12:27:21+03:00</updated_at>
<virtual_machine_id type="integer">1670</virtual_machine_id>
</load_balancing_cluster_node>
</nodes>
<ports type="array">
<port type="integer">80</port>
<port type="integer">345</port>
<port type="integer">678</port>
</ports>
<load_balancer>
<admin_note nil="true"/>
<allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
<allow_hot_migrate type="boolean">true</allow_hot_migrate>
<allow_swap type="boolean">true</allow_swap>
<booted type="boolean">true</booted>
<created_at type="datetime">2013-08-01T18:13:37+03:00</created_at>
<customer_network_id nil="true"/>
<deleted_at nil="true"/>
<edge_server_type nil="true"/>
<enable_autoscale nil="true"/>
<enable_monitis type="boolean">false</enable_monitis>
<firewall_notrack type="boolean">false</firewall_notrack>
<hostname>zaza</hostname>
<hypervisor_id type="integer">3</hypervisor_id>
<id type="integer">1654</id>
<identifier>pop7ba0j4imc7e</identifier>
<initial_root_password>Mvhn1gUjXpdS</initial_root_password>
<initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
<label>zaza_B</label>
<local_remote_access_ip_address>109.123.91.38</local_remote_access_ip_address>
<local_remote_access_port type="integer">5904</local_remote_access_port>
<locked type="boolean">false</locked>
<memory type="integer">512</memory>
<min_disk_size type="integer">5</min_disk_size>
<note nil="true"/>
<operating_system>linux</operating_system>
<operating_system_distro>lbva</operating_system_distro>
<preferred_hvs type="array"/>
<recovery_mode nil="true"/>
<remote_access_password>Y7wuNG1EpkZO</remote_access_password>
<service_password nil="true"/>
<state>new</state>
<storage_server_type nil="true"/>
<strict_virtual_machine_id nil="true"/>
<suspended type="boolean">false</suspended>
<template_id type="integer">10</template_id>
<template_label>Load Balancer Virtual Appliance</template_label>
<updated_at type="datetime">2013-08-01T18:43:01+03:00</updated_at>
<user_id type="integer">337</user_id>
<vip nil="true"/>
<xen_id type="integer">215</xen_id>

<ip_addresses type="array">
	<ip_address>
		<address>109.123.91.131</address>
		<broadcast>109.123.91.191</broadcast>
		<created_at type="datetime">2013-06-10T15:11:02+03:00</created_at>
		<customer_network_id nil="true"/>
		<disallowed_primary type="boolean">false</disallowed_primary>
		<gateway>105.123.91.129</gateway>
		<hypervisor_id nil="true"/>
		<id type="integer">10</id>
		<ip_address_pool_id nil="true"/>
		<network_address>109.123.91.128</network_address>
		<network_id type="integer">1</network_id>
		<pxe type="boolean">false</pxe>
		<updated_at type="datetime">2013-08-01T18:13:38+03:00</updated_at>
		<user_id nil="true"/>
		<free type="boolean">false</free>
		<netmask>255.255.255.192</netmask>
	</ip_address>
</ip_addresses>

<monthly_bandwidth_used type="decimal">36945.0</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>
<price_per_hour type="decimal">5250.0</price_per_hour>
<price_per_hour_powered_off type="decimal">2625.0</price_per_hour_powered_off>
<cpu_priority type="integer">10</cpu_priority>
</load_balancer>

<auto_scaling_out_memory>
<created_at type="datetime">2013-08-05T10:58:42+03:00</created_at>
Where:

- **cluster_type** – the type of the cluster (either cluster or autoscaleout)
- **config** – configuration array, where:
  - **max_node_amount** – maximum number of nodes (for autoscaling types; remains empty for cluster types)
  - **min_node_amount** – minimum number of nodes (for autoscaling types; remains empty for cluster types)
- **created_at** - the date when the cluster was created
- **id** – ID of the cluster
- **identifier** – the LB identifier in the DB
- **image_template_id** – the ID of a template on which the nodes of this load balancer are based (empty for cluster type)
- **load_balancer_id** - the ID of a load balancer added to this cluster
- **load_balancer_password** – root password, which is generated automatically
- **name** - load balancing cluster name
- **node_attributes** – an array of node attributes for autoscaling type, including cpu_shares, memory (RAM), rate_limit (port speed) and cpus (remains empty for cluster type)
  - **cpus** – the number of CPU cores allocated to this load balancer
- **cpu_shares** – the CPU priority of this load balancing cluster
- **memory** – the amount of RAM allocated to this load balancing cluster
- **rate_limit** - the port speed, set for the LB

**updated_at** – the date when the cluster was updated

**user_id** – ID of the load balancing cluster owner

**nodes** - an array of load balancing cluster nodes with VS details:
- **created_at** – the date when the cluster node was created
- **cluster_id** - the ID of load balancing cluster to which this node belongs
- **ip_address_id** – the ID of VS IP address added to a cluster
- **id** – node ID
- **updated_at** – the date when the cluster node was updated
- **virtual_machine_id** – the ID of VS added to a cluster

**ports** – the array of ports on which this cluster runs
- **port** – the cluster port

**load_balancer** - an array of load balancer details:
- **add_to_marketplace** – this parameter is not applicable to load balancers
- **admin_note** – an optional text note
- **allow_resize_without_reboot** – true if you can resize a VS's CPU and RAM without rebooting it
- **allowed_hot_migrate** – true if hot migration is allowed
- **allowed_swap** – true if swap disks are allowed, otherwise false
- **booted** - true if the server is booted, otherwise false
- **built** – true if the load balancing cluster is built, otherwise false
- **cpu_shares** – the CPU priority of this load balancing cluster
- **cpus** – the number of CPU cores allocated to this load balancer
- **created_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **customer_network_id** - this parameter is not applicable to load balancers
- **deleted_at** the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **edge_server_type** -this parameter is not applicable to load balancers
- **enable_autoscale** – true if autoscaling is enabled, otherwise false
- **firewall_notrack** - true if the NOTRACK rule is set in iptables
- **hostname** – the load balancer host name
- **hypervisor_id** – IDs of the compute resources used by this load balancing cluster
- **id** – the load balancing cluster ID
- **identifier** – identifier of the load balancer in the database
- **initial_root_password** — the LB root password
- **initial_root_password_encrypted** - true, if the root password is encrypted, otherwise false
- **label** – the load balancer name
- **local_remote_access_ip_address** - IP address used for remote access
- **local_remote_access_port** – the port ID used for console access
- **locked** – true if locked, otherwise false
- **memory** – the amount of RAM allocated to this load balancing cluster
- **min_disk_size** – the minimum disk size in GB required for a specified template
- **note** – an optional text, added as a note
- **operating_system** - the OS on which the load balancing cluster is based
- **operating_system_distro** – the distribution of the OS
- **preferred_hvs** - the array of preferable compute resources based on compute zone that meet some load balancer configuration settings
- **recovery_mode** – true if recovery mode is allowed, otherwise false
- **remote_access_password** – the password for remote access
- **service_password** – this parameter is not applicable to load balancers
- **state** – deprecated attribute
- **storage_server_type** – this parameter is not applicable to load balancers
- **strict_virtual_machine_id** – the ID of a VS that will never reside in this load balancing cluster
- **suspended** – true if suspended, otherwise false
- **template_id** – ID of the LB template
- **template_label** - the name of the template on which this load balancing cluster is based
- **updated_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **user_id** – the ID of the user who owns this load balancing cluster
- **vip** – true if the VIP status is set, otherwise false
- **xen_id** - the VS ID set by the virtualization engine
- **ip_addresses** - an array of IP addresses assigned to this load balancer and their details:
  - **address** – IP address
  - **broadcast** – broadcast address
  - **created_at** — the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - **customer_network_id** - the ID of the customer VLAN the IP address belongs to
  - **disallowed_primary** – true if not allowed to be used as primary, otherwise false
  - **gateway** – gateway address
  - **hypervisor_id** - the ID of a compute resource the IP address is associated with
  - **id** –the ID of the IP address
  - **ip_address_pool_id** - ID of the IP address pool the IP address is associated with
  - **network_address** – the address of the network
  - **network_id** –the ID of the network
  - **pxe** - true, if this address can be used for cloudbooting a compute resource
- **monthly_bandwidth_used** - LB monthly bandwidth
- **total_disk_size** - total LB disk size
- **price_per_hour** - price per hour set for this load balancer
- **price_per_hour_powered_off** - price per hour set for this load balancer in the powered off state
- **cpu_priority** - this is a new parameter reserved for future use; currently it has the same value as **cpu_shares** parameter

**auto_scaling_out_cpu** - an array of CPU autoscale out settings defining when the system should add more nodes to this autoscaling cluster:
- **created_at** - time when the CPU autoscale out settings were set
- **enabled** - true, if enabled, otherwise false
- **for_minutes** - the time threshold before scaling will be triggered
- **id** - parameter ID
- **units** - an amount of nodes that will be added when the **value** limit is reached
- **updated_at** - time when the CPU autoscale out settings were updated
- **value** - if the CPU usage is above percentage. An amount of nodes specified in **units** parameter will be added until the limit specified is reached.

**auto_scaling_in_cpu** - an array of CPU autoscale in settings:
- **created_at** - time when the CPU autoscale in settings were set
- **enabled** - true, if enabled, otherwise false
- **for_minutes** - the time threshold before scaling will be triggered
- **id** - parameter ID
- **units** - an amount of nodes that will be removed when the **value** limit is reached
- **updated_at** - time when the CPU autoscale in settings were updated
- **value** - if the CPU usage is less percentage. An amount of nodes specified in **units** parameter will be removed until the limit specified is reached.

**auto_scaling_in_memory** - an array of memory autoscale in settings:
- **created_at** - time when the memory autoscale in settings were set
- **enabled** - true, if enabled, otherwise false
- **for_minutes** - the time threshold before scaling will be triggered
- **id** - parameter ID
- **units** - an amount of nodes that will be removed when the **value** limit is reached
- **updated_at** - time when the memory autoscale in settings were updated
- **value** - if the memory usage is less percentage. An amount of nodes specified in **units** parameter will be removed until the limit specified is reached.
49.5 Get Load Balancer Billing Statistics

You can view the billing statistics for a particular load balancer using the following request:

GET /load_balancers/:load_balancer_id/vm_stats/:hourly_statistics_id.xml
GET /load_balancers/:load_balancer_id/vm_stats/hourly_statistics_id.json

Define a shorter period by setting Start and End time in the API call:

**XML Request Example**


**JSON Request example**


**XML Output Example**
<vm_stats>
  <created_at type="datetime">2013-05-02T06:00:27Z</created_at>
  <currency_code>USD</currency_code>
  <id type="integer">15490</id>
  <stat_time type="datetime">2013-05-02T06:00:00Z</stat_time>
  <updated_at type="datetime">2013-05-02T06:00:27Z</updated_at>
  <user_id type="integer">307</user_id>
  <virtual_machine_id type="integer">1214</virtual_machine_id>
  <vm_billing_stat_id type="integer">8089</vm_billing_stat_id>
  <billing_stats><disks type="array">
    <disk>
      <id type="integer">2430</id>
      <costs type="array">
        <cost>
          <value type="integer">100</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_min_iops</resource_name>
        </cost>
        <cost>
          <value type="integer">5</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_size</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_read</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_written</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>reads_completed</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>writes_completed</resource_name>
        </cost>
      </costs>
      <label nil="true"/>
    </disk>
    <disk>
      <id type="integer">2431</id>
      <costs type="array">
        <cost>
          <value type="integer">100</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_min_iops</resource_name>
        </cost>
        <cost>
          <value type="integer">1</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_size</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_read</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_written</resource_name>
        </cost>
      </costs>
    </disk>
  </billing_stats>
</vm_stats>
<cost>
  <value type="integer">0</value>
</cost>
<cost type="float">0.0</cost>
<resource_name>reads_completed</resource_name>
</cost>
<cost>
  <value type="integer">0</value>
</cost>
<cost type="float">0.0</cost>
<resource_name>writes_completed</resource_name>
</cost>
<cost>
  <value type="integer">0</value>
</cost>
<cost type="float">0.0</cost>
<resource_name>ip_addresses</resource_name>
</cost>
<cost>
  <value type="integer">1</value>
</cost>
<cost type="float">0.0</cost>
<resource_name>rate</resource_name>
</cost>
<cost>
  <value type="integer">0</value>
</cost>
<cost type="float">0.0</cost>
<resource_name>data_received</resource_name>
</cost>
<cost>
  <value type="integer">0</value>
</cost>
<cost type="float">0.0</cost>
<resource_name>data_sent</resource_name>
</cost>
<cost>
  <value type="integer">1</value>
</cost>
<cost type="float">0.0</cost>
<resource_name>eth0</resource_name>
</network_interface>
</network_interfaces>
<virtual_machines type="array">
  <virtual_machine>
    <id type="integer">1214</id>
    <costs type="array">
      <cost>
        <value type="integer">6</value>
      </cost>
      <cost type="float">0.0</cost>
      <resource_name>template</resource_name>
    </costs>
    <cost>
      <value type="integer">0</value>
    </cost>
    <cost type="float">0.0</cost>
    <resource_name>cpu_usage</resource_name>
  </virtual_machine>
  <label>OH-site</label>
</virtual_machines>
</billing_stats>
<total_cost type="float">0.0</total_cost>
<vm_resources_cost type="float">0.0</vm_resources_cost>
<usage_cost type="float">0.0</usage_cost>
</vm_stats>
Where:

created_at – the timestamp in DB when this record was created
updated_at – the time stamp in DB when this record was updated
currency_code - currency in which this load balancer is charged within the bucket
id – the ID of the load balancer hourly statistics
stat_time – the particular hour for which these statistics were generated
user_id - the ID of VS owner
virtual_machine_id - ID of a virtual server
virtual_machine_billing_statistics_id - ID of a load balancer billing statistics
billing_stats - an array of billing details for the resources used by this load balancer:

- disks - an array of disks used by this load balancer with their billing details:
  - label - disk name used in UI
  - id - disk ID used in database
  - costs- an array of disk related resources with their total prices for the period specified in the stat-time parameter, where:
    - resource_name - the resource in question. This can be disk_size, data_read, data_written, reads_completed and writes_completed
    - value - the amount of resources used (GBs of disk size, Kbs of data read/written, the number of reads/writes)
    - cost - the total due for the resource

- network_interfaces - an array of network interfaces used by this load balancer with their billing statistics:
  - label - network interface name used in OnApp
  - id - network interface ID
  - costs- an array of network interface related resources with their total prices for the period specified in the stat-time parameter, where:
    - resource_name - the resource in question. This can be ip_addresses, rate, data_received and data_sent
    - value - the amount of resources used by this network interface (the number of IPs, the port speed in Mb per second, the data sent and received in GB)
    - cost - the total due for the resource

- virtual_machines - an array of load balancer billing details:
  - label - load balancer name
  - costs- An array of load balancer resources with their total prices for the period specified in the stat-time parameter, where:
    - resource_name - the resource in question. This can be cpu_shares, cpus, memory, cpu_usage and template
    - value - the amount of resources allocated to this load balancer. For the templates resource, this parameter means a template ID in database.
    - cost - the total due for this resource
  - id - load balancer ID

- total_cost – the total amount of money owed for the load balancer specified by id parameter for a particular hour specified by stat_time parameter (total_cost = vm_resources_cost + usage_cost)
- `vm_resources_cost` – the amount of money due for the load balancer resources for the particular hour specified by `stat_time` parameter (memory, disks, templates)
- `usage_cost` – the total due for load balancer usage for this particular hour specified by `stat_time` parameter (data sent/received, bandwidth, CPU usage)

### 49.6 Get List of Load Balancer Autoscaling Monitors

To get details of RAM and CPU autoscaling monitors, use the following request:

GET /load_balancers/:load_balancer_id/monitis_monitors.xml
GET /load_balancers/:load_balancer_id/monitis_monitors.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**
<monitors type="array">
  <vm_monitor>
    <changed_at type="datetime">2013-07-31T12:38:56+03:00</changed_at>
    <created_at type="datetime">2013-07-31T12:38:56+03:00</created_at>
    <failures type="integer">0</failures>
    <id type="integer">7</id>
    <identifier>46469</identifier>
    <load_balancing_cluster_node_id type="integer">15</load_balancing_cluster_node_id>
    <name>cpu</name>
    <should_update type="boolean">false</should_update>
    <time>12:44</time>
    <updated_at type="datetime">2013-08-01T12:38:56+03:00</updated_at>
    <virtual_machine_id type="integer">1613</virtual_machine_id>
    <info>
      <id type="integer">46469</id>
      <kernelMax type="float">66.0</kernelMax>
      <tag>cluster-7b6282fb11da2686554e37186b42c24ff012647</tag>
      <agentPlatform>LINUX</agentPlatform>
      <name>yq1rlbcc8fx5sm cpu monitor</name>
      <niceMax type="float">66.0</niceMax>
      <agentKey>yq1rlbcc8fx5sm</agentKey>
      <userMax type="float">66.0</userMax>
      <ioWaitMax type="float">66.0</ioWaitMax>
      <idleMin type="float">0.0</idleMin>
      <ip>127.0.0.1</ip>
    </info>
    <stats type="array">
      <stat>
        <idleValue type="float">99.6</idleValue>
        <time>14:40</time>
        <userValue type="float">0.2</userValue>
        <status>OK</status>
        <ioWaitValue type="float">0.0</ioWaitValue>
        <kernelValue type="float">0.2</kernelValue>
        <niceValue type="float">0.0</niceValue>
        <cpuIndex type="integer">0</cpuIndex>
      </stat>
      <stat>
        <idleValue type="float">99.6</idleValue>
        <time>14:40</time>
        <userValue type="float">0.2</userValue>
        <status>OK</status>
        <ioWaitValue type="float">0.0</ioWaitValue>
        <kernelValue type="float">0.2</kernelValue>
        <niceValue type="float">0.0</niceValue>
        <cpuIndex type="integer">1</cpuIndex>
      </stat>
    </stats>
  </vm_monitor>
  <vm_monitor>
    <changed_at type="datetime">2013-07-31T13:15:09+03:00</changed_at>
    <created_at type="datetime">2013-07-31T13:15:09+03:00</created_at>
    <failures type="integer">0</failures>
    <id type="integer">8</id>
    <identifier>45671</identifier>
    <load_balancing_cluster_node_id type="integer">15</load_balancing_cluster_node_id>
    <name>memory</name>
    <should_update type="boolean">false</should_update>
    <time>15:09</time>
    <updated_at type="datetime">2013-07-31T15:11:52+03:00</updated_at>
    <virtual_machine_id type="integer">1613</virtual_machine_id>
    <info>
      <freeLimit type="float">200.0</freeLimit>
      <id type="integer">45671</id>
    </info>
  </vm_monitor>
</monitors>
<cachedLimit type="integer">1</cachedLimit>
<agentPlatform>Linux</agentPlatform>
<name>yq1rlbcc8fx5sm_memory_monitor</name>
<agentKey>yq1rlbcc8fx5sm</agentKey>
<checkInterval type="integer">100</checkInterval>
<bufferedLimit type="integer">-1</bufferedLimit>
<freeSwapLimit type="float">200.0</freeSwapLimit>
<ip>127.0.0.1</ip>

</info>
</stats>
</vm_monitor>
</monitors>

Where:

CPU monitor details:

vm_monitor - an array of load balancer autoscaling monitor details:

- **changed_at** - the time stamp when this record was updated
- **created_at** – the time stamp in DB when this record was created
- **updated_at** – the time stamp in DB when this record was updated
- **failures** - the number of detected failures
- **id** - monitis monitor ID
- **identifier** - monitis monitor identifier
- **load_balancing_cluster_node_id** - cluster node ID
- **name** - monitor name
- **time** - time when the monitor test was performed
- **virtual_machine_id** - ID of a load balancer on which the monitor was performed

info - an array of autoscaling monitor details:

- **id** - monitor ID
- **kernelMax** - maximum CPU value for kernel
- **tag** - CPU test tag
- **agentPlatform** - virtual server OS
- **name** – CPU test label
- **niceMax** - maximum CPU value for nice
- **agentKey** - virtual server identifier
- **userMax** - maximum CPU value for user processes
- **iowaitMax** - maximum CPU value for iowait
- **idleMin** - minimum CPU value for idle mode
- ip – virtual server IP address

stats - an array of statistics details:
  - idleValue - percentage of CPU used in idle mode
  - time - time when the statistics was gathered
  - userValue - percentage of CPU used in user mode
  - status - monitor status: OK, if the monitor is correct or NOK, if the autoscaling configuration does not match. Monitor status is refreshed once in 5 minutes.
  - ioWaitValue - percentage of time the CPU was idle during the IO request
  - kernelValue - percentage of CPU used by kernel
  - niceValue - percentage of CPU time occupied by processes with positive CPU value
  - cpuIndex - CPU number

Memory monitor details:

vm_monitor - an array of load balancer autoscaling monitor details:
  - changed_at - the time stamp when this record was updated
  - created_at - the time stamp in DB when this record was created
  - updated_at - the time stamp in DB when this record was updated
  - failures - the number of detected failures
  - id - monitis monitor ID
  - identifier - monitis monitor identifier
  - load_balancing_cluster_node_id - cluster node ID
  - name - monitor name
  - time - time when the monitor test was performed
  - virtual_machine_id - ID of a load balancer on which the monitor was performed

info - an array of monitor details:
  - freeLimit - free memory limit in MB
  - id - monitor ID
  - cachedLimit - cached memory limit in MB
  - agentPlatform - virtual server OS platform
  - name - test label
  - agentKey - virtual server identifier
  - checkInterval - monitor status refresh interval.
  - bufferedLimit - buffered memory limit in MB
  - freeSwapLimit - free swap limit in MB

stats - an array of statistics details
  - buffered - free virtual server memory in MB
• **cached** - cached virtual server memory in MB

• **freeswap** - free virtual server swap memory in MB

• **time** - time when the statistics was gathered.

• **totalMemory** - total virtual server memory in MB

• **status** - monitor status: OK, if the monitor is correct or NOK, if the autoscaling configuration does not match. Monitor status is refreshed once in 5 minutes

• **freeMemory** - free virtual server memory in MB

• **totalSwap** - total virtual server swap memory in MB

### 49.7 Get Load Balancer Autoscaling Monitor Details

To get details for a particular load balancer, use the following request:

**GET**

```
GET
/load_balancers/:load_balancer_id/monitis_monitors/:monitis_monitor_id.xml
GET
/load_balancers/:load_balancer_id/monitis_monitors/:monitis_monitor_id.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

Depending on the autoscaling monitor type - CPU or RAM, the output will be as follows:

**XML Output Example for CPU monitor**
<vm_monitor>
  <changed_at type="datetime">2013-07-31T12:38:56+03:00</changed_at>
  <created_at type="datetime">2013-07-31T12:38:56+03:00</created_at>
  <failures type="integer">0</failures>
  <id type="integer">7</id>
  <identifier>46469</identifier>
  <load_balancing_cluster_node_id type="integer">15</load_balancing_cluster_node_id>
  <name>cpu</name>
  <should_update type="boolean">false</should_update>
  <time>12:44</time>
  <updated_at type="datetime">2013-08-01T12:47:24+03:00</updated_at>
  <virtual_machine_id type="integer">1613</virtual_machine_id>
  <info>
    <id type="integer">46469</id>
    <kernelMax type="float">66.0</kernelMax>
    <tag>cluster-7b5282fb11da26865544e371865b24c4f012647</tag>
    <agentPlatform>LINUX</agentPlatform>
    <name>yq1rlbcc8fx5sm_cpu_monitor</name>
    <niceMax type="float">66.0</niceMax>
    <agentKey>yq1rlbcc8fx5sm</agentKey>
    <userMax type="float">66.0</userMax>
    <ioWaitMax type="float">66.0</ioWaitMax>
    <idleMin type="float">0.0</idleMin>
    <ip>127.0.0.1</ip>
  </info>
  <stats type="array">
    <stat>
      <idleValue type="float">99.6</idleValue>
      <time>14:40</time>
      <userValue type="float">0.2</userValue>
      <status>OK</status>
      <ioWaitValue type="float">0.0</ioWaitValue>
      <kernelValue type="float">0.2</kernelValue>
      <niceValue type="float">0.0</niceValue>
      <cpuIndex type="integer">0</cpuIndex>
    </stat>
    <stat>
      <idleValue type="float">99.6</idleValue>
      <time>14:40</time>
      <userValue type="float">0.2</userValue>
      <status>OK</status>
      <ioWaitValue type="float">0.0</ioWaitValue>
      <kernelValue type="float">0.2</kernelValue>
      <niceValue type="float">0.0</niceValue>
      <cpuIndex type="integer">1</cpuIndex>
    </stat>
  </stats>
</vm_monitor>

Where:

* **vm_monitor** - an array of load balancer autoscaling monitor details:
  
  - **changed_at** - the time stamp when this record was updated
  - **created_at** – the time stamp in DB when this record was created
  - **updated_at** – the time stamp in DB when this record was updated
  - **failures** - the number of detected failures
  - **id** - monitis monitor ID
- identifier - monitis monitor identifier
- load_balancing_cluster_node_id - cluster node ID
- name - monitor name
- time - time when the monitor test was performed
- virtual_machine_id - ID of a load balancer on which the monitor was performed

*info* - an array of autoscaling monitor details:
- id - monitor ID
- kernelMax - maximum CPU value for kernel
- tag - CPU test tag
- agentPlatform - virtual server OS
- name - CPU test label
- niceMax - maximum CPU value for nice
- agentKey - virtual server identifier
- userMax - maximum CPU value for user processes
- iowaitMax - maximum CPU value for iowait
- idleMin - minimum CPU value for idle mode
- ip - virtual server IP address

*stats* - an array of statistics details:
- idleValue - percentage of CPU used in idle mode
- time - time when the statistics was gathered
- userValue - percentage of CPU used in user mode
- status - monitor status: OK, if the monitor is correct or NOK, if the autoscaling configuration does not match. Monitor status is refreshed once in 5 minutes.
- ioWaitValue - percentage of time the CPU was idle during the IO request
- kernelValue - percentage of CPU used by kernel
- niceValue - percentage of CPU time occupied by processes with positive CPU value
- cpuIndex - CPU number

**XML Output Example for RAM monitor**
<vm_monitor>
  <changed_at type="datetime">2013-07-31T13:23:58+03:00</changed_at>
  <created_at type="datetime">2013-07-31T13:23:58+03:00</created_at>
  <failures type="integer">0</failures>
  <id type="integer">8</id>
  <identifier>45671</identifier>
  <load_balancing_cluster_node_id type="integer">15</load_balancing_cluster_node_id>
  <name>memory</name>
  <should_update type="boolean">false</should_update>
  <time>15:09</time>
  <updated_at type="datetime">2013-07-31T15:11:52+03:00</updated_at>
  <virtual_machine_id type="integer">1613</virtual_machine_id>
  <info>
    <freeLimit type="float">200.0</freeLimit>
    <id type="integer">45671</id>
    <cachedLimit type="integer">-1</cachedLimit>
    <agentPlatform>LINUX</agentPlatform>
    <name>yq1rlbcc8fx5sm_memory_monitor</name>
    <agentKey>yq1rlbcc8fx5sm</agentKey>
    <checkInterval type="integer">100</checkInterval>
    <bufferedLimit type="integer">-1</bufferedLimit>
    <freeSwapLimit type="float">200.0</freeSwapLimit>
    <ip>127.0.0.1</ip>
  </info>
  <stats>
    <buffered type="float">4.0</buffered>
    <cached type="float">34.0</cached>
    <freeswap type="float">1023.0</freeswap>
    <time>15:09</time>
    <totalMemory type="float">364.0</totalMemory>
    <status>OK</status>
    <freeMemory type="float">279.0</freeMemory>
    <totalSwap type="float">1023.0</totalSwap>
  </stats>
</vm_monitor>

Where:

vm_monitor - an array of load balancer autoscaling monitor details:

- changed_at - the time stamp when this record was updated
- created_at – the time stamp in DB when this record was created
- updated_at – the time stamp in DB when this record was updated
- failures - the number of detected failures
- id - monitis monitor ID
- identifier - monitis monitor identifier
- load_balancing_cluster_node_id - cluster node ID
- name - monitor name
- time - time when the monitor test was performed
- virtual_machine_id - ID of a load balancer on which the monitor was performed

info - an array of monitor details:

- freeLimit - free memory limit in MB
• **id** - monitor ID
• **cachedLimit** - cached memory limit in MB
• **agentPlatform** - virtual server OS platform
• **name** - test label
• **agentKey** - virtual server identifier
• **checkInterval** - monitor status refresh interval
• **bufferedLimit** - buffered memory limit in MB
• **freeSwapLimit** - free swap limit in MB

**stats** - an array of statistics details:
• **buffered** - free virtual server memory in MB
• **cached** - cached virtual server memory in MB
• **freeswap** - free virtual server swap memory in MB
• **time** - time when the statistics was gathered

• **totalMemory** - total virtual server memory in MB

• **status** - monitor status: OK, if the monitor is correct or NOK, if the autoscaling configuration does not match. Monitor status is refreshed once in 5 minutes

• **freeMemory** - free virtual server memory in MB
• **totalSwap** - total virtual server swap memory in MB

### 49.8 Add Load Balancing Cluster

To add a cluster type or an autoscaling type, use the following request:

**POST** /load_balancing_clusters.xml
**POST** /load_balancing_clusters.json

**XML Request Example**

```
curl -i -X POST -H 'Content-Type: application/xml' -H 'Accept:application/xml' -d '<load_balancing_cluster><ports type="array"><port>8080</port></ports><nodes_attributes type="array"><nodes_attribute><ip_address_id>5</ip_address_id><virtual_machine_id>1647</virtual_machine_id></nodes_attribute></nodes_attributes><cluster_type>cluster</cluster_type><load_balancer_attributes><label>cluster_xml</label><hypervisor_group_id>15</hypervisor_group_id><hypervisor_id>3</hypervisor_id><primary_network_group_id>4</primary_network_group_id><rate_limit>0</rate_limit><cpu_priority>10</cpu_priority><hostname>cluster_xml</hostname></load_balancer_attributes></load_balancing_cluster>' -u user:password http://onapp.test/load_balancing_clusters.xml
```

**JSON Request Example**
curl -i -X POST -H 'Content-Type: application/json' -H 'Accept: application/json' -d '{"load_balancing_cluster":{"ports":["8080"],"load_balancer_attributes":{"label":"test","hostname":"test","hypervisor_group_id":"15","hypervisor_id": "3","primary_network_group_id": "4","rate_limit": "1","cpu_priority": "10"},"cluster_type": "cluster","nodes_attributes": [{"ip_address_id": "5","virtual_ machine_id": "1647"}],"ports": [8000]}' -u user:password http://onapp.test/load_balancing_clusters.json

Where:

- **load_balancing_cluster** – an array with load balancing cluster details, where:
  - **load_balancer_attributes** – an array of LB instance, where:
    - **label** – the LB title
    - **hostname** – the host name of the load balancer
    - **hypervisor_group_id** – the ID of a compute zone
    - **hypervisor_id** – the ID of a compute resource
    - **primary_network_group_id** – the ID of a network zone assigned to the load balancer cluster
    - **rate_limit** – the port speed for the LB
    - **cpu_priority** – the CPU priority for the LB
  - **cluster_type** – the type of the load balancing cluster. Input *cluster* for the cluster type
  - **nodes_attributes** – an array of cluster nodes, where:
    - **virtual_machine_id** – the ID of virtual server, which is added as a node
    - **ip_address_id** – the ID of virtual server IP
  - **ports** – an array of ports on which an LB cluster will run
    - **port** – specify the port for this load balancer to run on (e.g. 9090, 8080, 9008, etc.)

Page History

v.6.1

- added the *cpu_priority* parameter

### 49.9 Add Autoscaling Cluster

To add an autoscaling cluster, use the following request:

- POST /load_balancing_clusters.xml
- POST /load_balancing_clusters.json

XML Request Example
curl -X POST -d
"<load_balancing_cluster><config><max_node_amount>4</max_node_amount><min_node_amount>2</min_node_amount><auto_scaling_in_cpu_attributes><for_minutes>20</for_minutes><units>1</units></auto_scaling_in_cpu_attributes><ports type="array"><port>80</port><port>25000</port></ports><auto_scaling_in_memory_attributes><for_minutes>20</for_minutes><units>1</units></auto_scaling_in_memory_attributes><auto_scaling_out_memory_attributes><for_minutes>5</for_minutes><units>1</units></auto_scaling_out_memory_attributes><load_balancer_attributes><label>test</label><hostname>aa</hostname><rate_limit>0</rate_limit><primary_network_group_id>3</primary_network_group_id><hypervisor_group_id>1</hypervisor_group_id><hypervisor_id>1</hypervisor_id><cpu_priority>10</cpu_priority></load_balancer_attributes><cluster_type>autoscaleout</cluster_type><node_attributes><cpus>1</cpus><cpu_shares>1</cpu_shares><memory>128</memory><rate_limit>0</rate_limit></node_attributes><auto_scaling_out_cpu_attributes><for_minutes>5</for_minutes><units>1</units></auto_scaling_out_cpu_attributes><image_template_id>62</image_template_id></load_balancing_cluster>" -u user:userpass http://onapp.test/load_balancing_clusters.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request Example**

curl -X POST -d
{"load_balancing_cluster":{"config":{"max_node_amount":"4","min_node_amount":"2"},"auto_scaling_in_cpu_attributes":{"for_minutes":"20","units":"1"},"auto_scaling_in_memory_attributes":{"for_minutes":"20","units":"1"},"auto_scaling_out_memory_attributes":{"for_minutes":"5","units":"1"},"load_balancer_attributes":{"label":"test","hostname":"aa","rate_limit":0,"primary_network_group_id":3,"hypervisor_group_id":1,"hypervisor_id":10,"cluster_type":autoscaleout,"node_attributes":{"cpus":1,"cpu_shares":1,"memory":128,"rate_limit":0},"auto_scaling_out_cpu_attributes":{"for_minutes":5,"units":1,"enabled":true,"value":80},"image_template_id":62},"available_vms":null"} -u user:userpass http://onapp.test/load_balancing_clusters.json -H 'Accept: application/json' -H 'Content-type: application/json'

**Where:**

**Autoscaling cluster parameters:**

- `load_balancing_cluster` - an array with load balancing cluster details, where:
  - `config` - a configuration array, where:
    - `max_node_amount` - the maximum number of nodes in this cluster
    - `min_node_amount` - the minimum number of nodes in this cluster
  - `ports` - the array of ports on which a load balancing cluster will run
  - `load_balancer_attributes` - an array of LB instance, where:
    - `label` - the LB title
    - `rate_limit` - the port speed for the LB
    - `hostname` - the hostname of the load balancer
    - `primary_network_group_id` - the ID of a network zone assigned to the load balancer cluster
    - `hypervisor_group_id` - the ID of a compute zone
o **hypervisor_id** – the ID of a compute resource
o **cpu_cpriority** - the CPU priority of this load balancer

- **cluster_type** - type of load balancing cluster. Input autoscaleout for the autoscaling type
- **node_attributes** - an array of cluster nodes, where:
  o **cpus** - number of CPUs for each node
  o **cpu_shares** - the CPU priority of each node
  o **memory** - the amount of RAM for each node
  o **rate_limit** - the port speed for each node

- **auto_scaling_in_memory_attributes** - an array of RAM scale in attributes, where:
  o **for_minutes** - how long the RAM should be monitored. The **for_minutes** parameter must be divisible by 5.
  o **units** - how many nodes are removed from the cluster, if the rule is met
  o **enabled** - set 1/true if the rule is enabled. Otherwise set 0/false
  o **value** - the amount of RAM (MB). If this value is reached by the cluster during the period specified by the **for_minutes** parameter, the system will remove the amount of units set by the units parameters.

- **auto_scaling_in_cpu_attributes** - an array of CPU scale in attributes, similar to RAM scale in attributes

- **auto_scaling_out_memory_attributes** - an array of RAM scale out attributes, where:
  o **for_minutes** - how long the RAM should be monitored
  o **units** - how many nodes are added to the cluster if the rule is met
  o **enabled** - set 1/true to enable the rule. Otherwise set false/0.
  o **value** - the amount of RAM (MB). If this value is reached by the cluster during the period specified by the **for_minutes** parameter, the system will add the amount of units set by the units parameters.

- **auto_scaling_out_cpu_attributes** - an array of CPU scale out attributes, similar to RAM scale out attributes

**Page History**

v.6.1
- added the **cpu_priority** parameter

**49.10 Add Nodes to Cluster Type**

To add new VSs (nodes) to a cluster type, use the following request:

```
PUT /load_balancing_clusters/:id.xml
PUT /load_balancing_clusters/:id.json
```

**XML Request Example**
OnApp Cloud 6.3 API Guide

**curl**

```
curl -i -X PUT -H 'Content-Type: application/xml' -H 'Accept: application/xml' -d 'load_balancing_cluster<nodes_attributes type="array"><node_attribute><virtual_machine_id>1647</virtual_machine_id><ip_address_id>5</ip_address_id></node_attribute><node_attribute><virtual_machine_id>1520</virtual_machine_id><ip_address_id>9</ip_address_id></node_attribute></nodes_attributes></load_balancing_cluster>' -u user:password
http://onapp.test/load_balancing_clusters/22.xml
```

**JSON Request Example**

```
curl -i -X PUT -H 'Content-Type: application/json' -H 'Accept: application/json' -d 'load_balancing_cluster:nodes_attributes:[{virtual_machine_id:"1647",ip_address_id:"5"},{virtual_machine_id:"1520",ip_address_id:"9"}]'} -u user:password http://onapp.test/load_balancing_clusters/22.json
```

**Where:**

*node_attributes*– an array where you may add new nodes
-  *virtual_machine_id* - input the ID of the virtual server
-  *ip_address_id* - the ID of virtual server IP

**49.11 Remove Nodes from Cluster Type**

To remove nodes from cluster type, use the following request:

PUT /load_balancing_clusters/:id.xml
PUT /load_balancing_clusters/:id.json

**XML Request Example**

```
curl -i -X PUT -H 'Content-Type: application/xml' -H 'Accept: application/xml' -d 'load_balancing_cluster<nodes_attributes type="array"><node_attribute><_destroy>true</_destroy><id>28</id></node_attribute><node_attribute><_destroy>true</_destroy><id>29</id></node_attribute></nodes_attributes></load_balancing_cluster>' -u user:password
http://onapp.test/load_balancing_clusters/22.xml
```

**JSON Request Example**

```
curl -i -X PUT -H 'Content-Type: application/json' -H 'Accept: application/json' -d 'load_balancing_cluster:nodes_attributes:[{_destroy:"true","id":"26"},{_destroy:"true","id":"27"}]'} -u user:password
http://onapp.test/load_balancing_clusters/22.json
```

**Where:**

*load_balancing_cluster* – an array with load balancing cluster details:
-  *nodes_attributes*– an array where you may remove node
  -  *destroy* – set 1/true to remove this node from the cluster
49.12 Edit Load Balancing Cluster

To edit a particular load balancing cluster parameters, use the following request:

PUT /load_balancing_clusters/:id.xml
PUT /load_balancing_clusters/:id.json

XML Request Example

```
```

JSON Request Example

```
```

Where:
- `ports*` - edit ports on which the load balancing cluster runs
- `load_balancer_attributes*` - an array of LB instance, where:
  - `label*` - the LB title
  - `port_speed*` - the port speed for the LB
  - `cpu_priority*` - the CPU priority for the LB

Using this request you can edit the following load balancing cluster parameters: port, label, cpu priority, rate limit and add node to the load balancing cluster. To see how to add and remove nodes from the cluster type, see the Remove Nodes from Cluster Type and Add Nodes to Cluster Type sections.

Page History

v.6.1
- added the `cpu_priority` parameter

49.13 Edit Autoscaling Cluster

You may change minimum/maximum number of nodes of autoscaling type, as well as change the autoscaling attributes for RAM and CPU.

To configure autoscaling type, use the following request:

PUT /load_balancing_clusters/:id.xml
PUT /load_balancing_clusters/:id.json
**XML Request Example**

```xml
<load_balancing_cluster>
  <config>
    <max_node_amount>4</max_node_amount>
    <min_node_amount>2</min_node_amount>
  </config>
  <auto_scaling_in_cpu_attributes>
    <for_minutes>20</for_minutes>
    <units>1</units>
    <enabled>true</enabled>
    <value>60</value>
  </auto_scaling_in_cpu_attributes>
  <ports type="array">
    <port>80</port>
    <port>25000</port>
  </ports>
  <auto_scaling_in_memory_attributes>
    <for_minutes>20</for_minutes>
    <units>1</units>
    <enabled>true</enabled>
    <value>200</value>
  </auto_scaling_in_memory_attributes>
  <auto_scaling_out_memory_attributes>
    <for_minutes>5</for_minutes>
    <units>1</units>
    <enabled>true</enabled>
    <value>100</value>
  </auto_scaling_out_memory_attributes>
  <load_balancer_attributes>
    <label>az_A5</label>
    <rate_limit>0</rate_limit>
    <cpu_priority>10</cpu_priority>
  </load_balancer_attributes>
  <node_attributes>
    <cpus>1</cpus>
    <cpu_shares>1</cpu_shares>
    <memory>128</memory>
  </node_attributes>
  <auto_scaling_out_cpu_attributes>
    <for_minutes>5</for_minutes>
    <units>1</units>
    <enabled>true</enabled>
    <value>80</value>
  </auto_scaling_out_cpu_attributes>
</load_balancing_cluster>
```

**JSON Request Example**

```json
{
  "load_balancing_cluster": {
    "config": {
      "max_node_amount": "4",
      "min_node_amount": "2"
    },
    "auto_scaling_in_cpu_attributes": {
      "for_minutes": "20",
      "units": "1",
      "enabled": "true",
      "value": "60"
    },
    "ports": ["80", "25000"],
    "auto_scaling_in_memory_attributes": {
      "for_minutes": "20",
      "units": "1",
      "enabled": "true",
      "value": "200"
    },
    "auto_scaling_out_memory_attributes": {
      "for_minutes": "5",
      "units": "1",
      "enabled": "true",
      "value": "100"
    },
    "load_balancer_attributes": {
      "label": "az_A5",
      "rate_limit": 0,
      "cpu_priority": 10
    },
    "node_attributes": {
      "cpus": 1,
      "cpu_shares": 1,
      "memory": 128
    },
    "auto_scaling_out_cpu_attributes": {
      "for_minutes": "5",
      "units": "1",
      "enabled": "true",
      "value": "80"
    }
  }
}
```

Where:

- `load_balancing_cluster` - an array with load balancing cluster details, where:
  - `max_node_amount` – maximum number of nodes for the cluster
  - `min_node_amount` – minimum number of nodes for the cluster
  - `config` - a configuration array, where:
    - `max_node_amount` - the maximum number of nodes in this cluster
    - `min_node_amount` - the minimum number of nodes in this cluster
  - `auto_scaling_in_cpu_attributes` - an array of CPU scale in attributes, similar to RAM scale in attributes
    - `for_minutes` - how long the resource should be monitored. The for_minutes parameter must be divisible by 5.
    - `units` - how many nodes are removed or added to the cluster, if the rule is met
    - `enabled` - set 1/true if the rule is enabled. Otherwise set 0/false
    - `value` - the amount of resource. If this value is reached by the cluster during the period specified by the for_minutes parameter, the system will remove the amount of units set by the units parameter.
• **ports** – the array of ports on which a load balancing cluster will run

• **auto_scaling_in_memory_attributes** - an array of RAM scale in attributes, where:
  
  o **for_minutes** - how long the RAM should be monitored. The **for_minutes** parameter must be divisible by 5.
  
  o **units** - how many nodes are removed from the cluster, if the rule is met
  
  o **enabled** - set 1/true if the rule is enabled. Otherwise set 0/false
  
  o **value** - the amount of RAM (MB). If this value is reached by the cluster during the period specified by the **for_minutes** parameter, the system will remove the amount of units set by the units parameters.

• **auto_scaling_out_memory_attributes** - an array of RAM scale out attributes, where:
  
  o **for_minutes** - how long the RAM should be monitored
  
  o **units** - how many nodes are added to the cluster if the rule is met
  
  o **enabled** - set 1/true to enable the rule. Otherwise set false/0.
  
  o **value** - the amount of RAM (MB). If this value is reached by the cluster during the period specified by the **for_minutes** parameter, the system will add the amount of units set by the units parameters.

• **load_balancer_attributes** - an array of LB instance, where:
  
  o **label** – the LB title
  
  o **rate_limit** - the port speed for the LB
  
  o **hostname** - the hostname of the load balancer
  
  o **hypervisor_id** – the ID of a compute resource
  
  o **cpu_priority** – the CPU priority for the LB

• **cluster_type** - type of load balancing cluster. Input **autoscaleout** for the autoscaling type

• **node_attributes** - an array of cluster nodes, where:
  
  o **cpus** - number of CPUs for each node
  
  o **cpu_shares** - the CPU priority of each node
  
  o **memory** - the amount of RAM for each node
  
  o **rate_limit** - the port speed for each node

• **auto_scaling_out_cpu_attributes** - an array of CPU scale out attributes, similar to RAM scale out attributes

---

### 49.14 Edit Load Balancing Cluster Ports

To set the list of ports on which a load balancing cluster runs, use the following request:

```
PUT /load_balancing_clusters/:id.xml
PUT /load_balancing_clusters/:id.json
```

**XML Request Example**

```
```
### JSON Request Example

```
```

Where:
- **ports** - an array of ports on which a load balancing cluster will run
- **port** - a particular port

#### 49.15 Delete Load Balancing Cluster

To delete a load balancing cluster, use the following request:

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/load_balancers/24.xml
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/load_balancers/24.json
```

Where you have to specify ID of a load balancer you want to delete.

#### 49.16 Rebuild Load Balancer

To rebuild a load balancer, use the following request:

**XML Request Example**

```
```

**JSON Request Example**
49.17 Search Load Balancer by Label

To search load balancer by label, use the following request:

GET http://onapp.test/load_balancers.xml?q=label
GET http://onapp.test/load_balancers.json?q=label

XML Request Example


JSON Request Example


Where you have to specify the label of a load balancer you are searching for.

49.18 Start up Load Balancer

To start up a load balancer, use the following request:

POST onapp.test/load_balancers/:load_balancer_id/startup.xml
POST onapp.test/load_balancers/:load_balancer_id/startup.json

XML Request Example


JSON Request Example


49.19 Shut Down Load Balancer

To shut down a load balancer, use the following request:
POST /load_balancers/:load_balancer_id/shutdown.xml
POST /load_balancers/:load_balancer_id/shutdown.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

### 49.20 Suspend Load Balancer

To suspend a load balancer, use the following request:

POST /load_balancers/:load_balancer_id/suspend.xml
POST /load_balancers/:load_balancer_id/suspend.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

To unsuspend a load balancer, use the same request again.

### 49.21 Stop Load Balancer

To stop a load balancer, use the following request:

POST /load_balancers/:load_balancer_id/stop.xml
POST /load_balancers/:load_balancer_id/stop.json

**XML Request Example**

```bash
```
49.22 Unlock Load Balancer

To unlock a load balancer, use the following request:
POST /load_balancers/:load_balancer_id/unlock.xml
POST /load_balancers/:load_balancer_id/unlock.json

XML Request Example


JSON Request Example

50 Locales

You can see the list of locales that you can assign to a user or a group of users. The array will contain the name of locale and its code.

To see the list of locales, use the following request:

GET /settings/locales.xml
GET /settings/locales.json

XML Request Example


JSON Request Example


XML Output example

```
<language_locales type="array">
  <language_locale>
    <id type="integer">1</id>
    <name>en</name>
    <created_at type="dateTime">2018-01-15T11:45:14+00:00</created_at>
    <updated_at type="dateTime">2018-01-15T11:45:14+00:00</updated_at>
  </language_locale>
  ...
  <language_locale></language_locale>
  ...
</language_locales>
```

Where:

- id - the code of the locale
- name - the name of the language

Page History

v.5.9

- removed the deprecated GET /settings/internationalization.xml and GET /settings/internationalization.json requests
51 Location Groups

Location groups allow manage the Compute resource, Data store, Backup server and Network zones in geographically dispersed locations in the same cloud. Currently, this enables you to host CDN Edge Servers and Storage Servers in remote locations using a single Control panel. The multi-location option is set in OnApp Dashboard for the cloud and its availability depends on the license type.

51.1 Get List of Location Groups

To get an array of location groups set up within your cloud, use the following request:

GET /settings/location_groups.xml
GET /settings/location_groups.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<location_groups type="array">
  <location_group>
    <id type="integer">1</id>
    <created_at type="dateTime">2016-05-06T18:35:46+03:00</created_at>
    <updated_at type="dateTime">2016-12-14T13:22:13+03:00</updated_at>
    <country>United States</country>
    <city>Alexandria</city>
    <federation_id>resource:compute:onapp-K1BrvBV19PYtxZeb2STgml3ihzk</federation_id>
    <lat type="float">38.8048</lat>
    <lng type="float">-77.0469</lng>
    <cdn_enabled type="boolean">false</cdn_enabled>
    <federated type="boolean">true</federated>
  </location_group>
  <location_group>
    <id type="integer">2</id>
    <created_at type="dateTime">2016-05-06T18:36:18+03:00</created_at>
    <updated_at type="dateTime">2016-12-14T13:22:14+03:00</updated_at>
    <country>United Kingdom</country>
    <city>London</city>
    <federation_id>resource:compute:onapp-Jtv9%2FAUtqM8mb0916Y08EvE0E</federation_id>
    <lat type="float">51.5074</lat>
    <lng type="float">-0.127758</lng>
    <cdn_enabled type="boolean">false</cdn_enabled>
    <federated type="boolean">true</federated>
  </location_group>
</location_groups>

Where:

- **id** - the location group ID
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** - the date when the location group was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **country** - the location group country
- **city** - the location group city
- **lat** - Latitude of the city
- **lng** - Longitude of the city
- **cdn_enabled** - true if CDN is enabled for the location group; otherwise false
- **federated** - true if the location group is federated; otherwise false

### 51.2 Get Location Group Details

The following method returns details for a particular location group:

GET /settings/location_groups/:id.xml
GET /settings/location_groups/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```
XML Output Example

```xml
<location_group>
    <city>Lviv</city>
    <country>Ukraine</country>
    <created_at type="datetime">2015-08-18T13:33:50+03:00</created_at>
    <federation_id nil="true"/>
    <id type="integer">3</id>
    <updated_at type="datetime">2015-08-18T13:33:50+03:00</updated_at>
    <cdn_enabled type="boolean">true</cdn_enabled>
    <federated type="boolean">false</federated>
</location_group>
```

Where:

- `created_at` - the date in the [YYYY]MMDD[T[hh][mm][ss]Z format
- `updated_at` - the date when the location group was updated in the [YYYY]MMDD[T[hh][mm][ss]Z format
- `city` - the location group City
- `country` - the location group country
- `id` - the location group ID
- `cdn_enabled` - true if CDN is enabled for the location group; otherwise false
- `federated` - true if the location group is federated; otherwise false

51.3 Refresh Location Groups

To refresh location groups and synchronize the groups in the Control Panel and the Dashboard, use the following request:

GET /settings/location_groups/refresh.xml
GET /settings/location_groups/refresh.json

XML Request Example

```bash
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass -X GET http://onapp.com/settings/location_groups/refresh.json
```
51.4 Attach Compute Zone to Location Group

To attach a compute zone to a location group, use the following request:

POST
/settings/location_groups/:location_group_id/hypervisor_groups/attach_resource.xml
POST
/settings/location_groups/:location_group_id/hypervisor_groups/attach_resource.json

XML Request Example

curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<attach_resource_id>1</attach_resource_id>' -url http://onapp.test/settings/location_groups/1/hypervisor_groups/attach_resource.xml

JSON Request Example


This request attaches a particular compute zone (*attach_resource_id*) to a specific location group (*:location_group_id*)

51.5 Detach Compute Zone from Location Group

To detach a compute zone from a location group, use the following request:

POST
/settings/location_groups/:location_group_id/hypervisor_groups/:hypervisor_group_id/detach_resource.xml
POST
/settings/location_groups/:location_group_id/hypervisor_groups/:hypervisor_group_id/detach_resource.json

XML Request Example


JSON Request Example

Returns HTTP 204 response on successful deletion, or HTTP 404 when a compute zone with the ID specified is not found, or the URL requested is incorrect.

51.6 Attach Data Store Zone to Location Group

To attach a data store zone to location group, use the following request:

POST 
/settings/location_groups/:location_group_id/data_store_groups/attach_resource.xml

XML Request Example


JSON Request Example


This request attaches a particular data store zone (attach_resource_id) to a specific location group (:location_group_id)

51.7 Detach Data Store Zone from Location Group

To detach a data store zone from a location group, use the following request:

POST 
/settings/location_groups/:location_group_id/data_store_groups/:data_store_group_id/detach_resource.xml

XML Request Example


JSON Request Example

51.8 Attach Network Zone to Location Group

To attach a network zone to a location group, use the following request:

```sh
POST /settings/location_groups/:location_group_id/network_groups/attach_resource.xml
POST /settings/location_groups/:location_group_id/network_groups/attach_resource.json
```

**XML Request Example**

```sh
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<attach_resource_id>1</attach_resource_id>' --url http://onapp.test/settings/location_groups/1/network_groups/attach_resource.xml
```

**JSON Request Example**

```sh
```

This request attaches a particular network zone (attach_resource_id*) to a specific location group (:location_group_id)

51.9 Detach Network Zone from Location Group

To detach a network zone from a location group, use the following request:

```sh
POST /settings/location_groups/:location_group_id/network_groups/:network_group_id/detach_resource.xml
POST /settings/location_groups/:location_group_id/network_groups/:network_group_id/detach_resource.json
```

**XML Request Example**

```sh
```

Returns HTTP/1.1 302 Found response on successful deletion, or HTTP 404 when a data store zone with the ID specified is not found, or the URL requested is incorrect.
51.10 Attach Backup Server Zone to Location Group

To attach a backup server zone to a location group, use the following request:

```
POST /settings/location_groups/:location_group_id/backup_server_groups/attach_resource.xml
```

**XML Request Example**

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d "" --url http://onapp.test/settings/location_groups/1/backup_server_groups/attach_resource.xml
```

**JSON Request Example**

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"attach_resource_id":"4"}' --url http://onapp.test/settings/location_groups/1/backup_server_groups/attach_resource.json
```

This request attaches a particular backup server zone (attach_resource_id) to a specific location group (:location_group_id)

51.11 Detach Backup Server Zone from Location Group

To detach a backup server zone from a location group, use the following request:

```
POST /settings/location_groups/:location_group_id/backup_server_groups/:backup_server_group_id/detach_resource.xml
```

Returns HTTP/1.1 302 Found response on successful deletion, or HTTP 404 when a network zone with the ID specified is not found, or the URL requested is incorrect.
OnApp Cloud 6.3 API Guide

/settings/location_groups/:location_group_id/backup_server_groups/:backup_server_group_id/detach_resource.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Returns HTTP/1.1 302 Found response on successful deletion, or HTTP 404 when a backup server zone with the ID specified is not found, or the URL requested is incorrect.

### 51.12 Get List of Compute Zones Attached to Location Group

To get the list of compute zones attached to a location group, use the following request:

GET /settings/location_groups/:location_group_id/hypervisor_groups.xml

GET /settings/location_groups/:location_group_id/hypervisor_groups.json

**XML Request Example**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/location_groups/1/hypervisor_groups.xml
```

**JSON Request Example**

```
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/location_groups/1/hypervisor_groups.json
```

**XML Output Example**
```xml
<hypervisor_groups type="array">
  <hypervisor_group>
    <closed type="boolean">false</closed>
    <created_at type="datetime">2013-09-04T12:49:45+03:00</created_at>
    <federation_enabled type="boolean">false</federation_enabled>
    <federation_id nil="true"/>
    <hypervisor_id nil="true"/>
    <id type="integer">1</id>
    <identifier nil="true"/>
    <label>KVM C5 Compute Zone</label>
    <location_group_id type="integer">1</location_group_id>
    <server_type>virtual</server_type>
    <traded type="boolean">false</traded>
    <updated_at type="datetime">2015-04-02T16:47:37+03:00</updated_at>
    <max_host_free_memory type="integer">3819</max_host_free_memory>
    <max_host_cpu type="integer">4</max_host_cpu>
    <prefer_local_reads type="boolean">false</prefer_local_reads>
    <vlan nil="true"/>
    <release_resource_type>ballooning</release_resource_type>
    <network_failure type="boolean">false</network_failure>
    <storage_channel type="integer">2</storage_channel>
    <run_sysprep type="boolean">true</run_sysprep>
    <default_gateway nil="true"/>
    <recovery_type>roundrobin</recovery_type>
    <failover_timeout type="integer">15</failover_timeout>
    <cpu_units type="integer">1000</cpu_units>
    <supplier_version nil="true"/>
    <supplier_provider nil="true"/>
  </hypervisor_group>
  ...
</hypervisor_groups>
```

**Where:**

- `created_at` - the date in the `[YYYY][MM][DD][T][hh][mm][ss]Z` format
- `federation_enabled` - not relevant to compute zones
- `federation_id` - not relevant to compute zones
- `closed` - not relevant to compute zones
- `traded` - true, if the zone came from the Federation and was subscribed to by the user
- `updated_at` - the date when the compute zone was updated in the `[YYYY][MM][DD][T][hh][mm][ss]Z` format
- `default_gateway` - external gateway IP address used for the VMware utilization with the external firewall. All virtual machines within a compute zone will be rerouted to this gateway
- `vlan` - address of a VLAN the default gateway is located on.
- `id` - the compute zone ID
- `label` - title of a compute zone
- `location_group_id` - ID of a location group the compute zone is assigned to
- `max_vms_start_at_once` - the maximum number of virtual servers that can be started simultaneously within this compute zone
- `network_failure` - true, if all compute resources in the compute zone failed
- `prefer_local_reads` - set 1 to minimize the network throughput dependency for read heavy workloads. When this option is enabled, reads go over the local software bridge to a local replica of the data rather than traverse a physical NIC + switch.
recovery_type - specify the compute resource selection algorithm, which will be used on virtual server provisioning and recovery, per compute zone:

- **roundrobin** - set the roundrobin type to select the compute resource with maximum free RAM during the VS recovery

  Note: this option behaves in different ways, depending on the event:
  - On provisioning, the round-robin algorithm will be used on compute resource selection.
  - On recovery, the compute resource with maximum free RAM will be selected.

- **fillnext** - select the fillnext type to select the compute resource with minimum required free RAM. This option allows to fill compute resource as tightly as possible before starting to use next appliance in the zone

release_resource_type - specify the release resource type. Release resource option allows to free up compute resource resources by over-committing RAM, CPU and CPU shares of virtual servers that are shut down.

- **memory_guarantee** - the actual free compute resource memory is calculated. All virtual servers residing on the compute resource will be able to start.

- **ballooning** - free compute resource memory is calculated with the ability to use memory over-committing. The ballooning option is only available for KVM compute resources. NOTE: Virtual server may be migrated to another compute resource if there is not enough memory for it to start up on the compute resource with the ballooning option enabled.

  Do not use the ballooning option if there is at least one edge or storage server within the compute zone.

- **only_started_vms** - only the free memory of running virtual servers is calculated.

  By default, the compute zone is created with the Memory Guarantee option enabled. In this case the release resources option is not used. Then, to enable resource over-committing you should choose either the Ballooning or Only Started VS option.

failover_timeout - time period for which the iterations will run during the failover

run_sysprep - set 1 to enable Windows virtual server deployment without running sysprep

server_type - specify the type of servers that will reside within this compute zone:

- **virtual** - choose the virtual type to create a Xen, KVM, VMware or CloudBoot zone

- **smart** - choose the smart server type to create a smart server zone

- **baremetal** - choose the baremetal server type to create a baremetal server zone
storage_channel - storage channel for the communication with the
max_host_free_memory - compute resource with maximum RAM value in this zone
max_host_cpu - compute resource with maximum RAM value in this zone

51.13 Get List of Data Store Zones Attached to Location Group

To get the list of data store zones attached to a location group, use the following request:

GET /settings/location_groups/:location_group_id/data_store_groups.xml
GET /settings/location_groups/:location_group_id/data_store_groups.json

XML Request Example

curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password --url http://onapp.test/settings/location_groups/12/data_store_groups.xml

JSON Request Example

curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:password --url http://onapp.test/settings/location_groups/12/data_store_groups.json

XML Output Example

<data_store_group>
  <closed type="boolean">false</closed>
  <created_at type="datetime">2013-06-10T15:09:58+03:00</created_at>
  <federation_enabled type="boolean">false</federation_enabled>
  <hypervisor_id nil="true"/>
  <id type="integer">86</id>
  <identifier nil="true"/>
  <label>DSZ</label>
  <location_group_id type="integer">12</location_group_id>
  <traded type="boolean">false</traded>
  <updated_at type="datetime">2013-06-10T15:09:58+03:00</updated_at>
  <default_burst_iops type="integer">15000</default_burst_iops>
  <default_max_iops type="integer">15000</default_max_iops>
  <min_disk_size type="integer">0</min_disk_size>
</data_store_group>

Where:
closed - not relevant to datastore zones
created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
federation_enabled - not relevant to datastore zones
federation_id - not relevant to datastore zones
hypervisor_id - ID of the compute resource the data store zone is attached to
id - the data store zone ID
identifier - identifier of the data store zone
**label** - the data store zone title

**location_group_id** - ID of a location group the data store zone is assigned to

**traded** - true, if the zone came from the Federation and was subscribed to by the user

**updated_at** - the date when the Data store zone was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**SolidFire related parameters:**

**default_burst_iops** - the default peak IOPS value

**default_max_iops** - the default maximum IOPS value for the data store zone

**min_disk_size** - minimum disk size for the data store zone

### 51.14 Get List of Network Zones Attached to Location Group

To get the list of network zones attached to a location group, use the following request:

GET /settings/location_groups/:location_group_id/network_groups.xml
GET /settings/location_groups/:location_group_id/network_groups.json

**XML Request Example**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/location_groups/1/network_groups.xml
```

**JSON Request Example**

```
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/location_groups/1/network_groups.json
```

**XML Output Example**

```
<network_groups type="array">
  <network_group>
    <closed type="boolean">false</closed>
    <created_at type="datetime">2013-09-04T12:51:11+03:00</created_at>
    <federation_enabled type="boolean">false</federation_enabled>
    <federation_id nil="true"/>
    <hypervisor_id nil="true"/>
    <id type="integer">5</id>
    <identifier nil="true"/>
    <label>Network Zone 1</label>
    <location_group_id type="integer">1</location_group_id>
    <traded type="boolean">false</traded>
    <updated_at type="datetime">2014-08-11T12:06:11+03:00</updated_at>
  </network_group>
...
</network_groups>
```

Where:

**federation_enabled** - not relevant to network zones

**federation_id** - not relevant to network zones
**closed** - not relevant to network zones

**traded** - true, if the zone came from the Federation and was subscribed to by the user

**label** - the network zone title

**location_group_id** - ID of a location group the network zone is assigned to

**created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**updated_at** - the date when the Network zone was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**id** - the network zone ID

**vlan** - the VLAN this network zone belongs to

### 51.15 Get List of Backup Server Zones Attached to Location Group

To get the list of backup server zones attached to a location group, use the following request:

**GET**

/settings/location_groups/:location_group_id/backup_server_groups.xml

**GET**

/settings/location_groups/:location_group_id/backup_server_groups.json

**XML Request Example**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/settings/location_groups/1/backup_server_groups.xml
```

**JSON Request Example**

```
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/location_groups/1/backup_server_groups.json
```

**XML Output Example**

```
<backup_server_groups>
<backup_server_group>
  <created_at type="datetime">2013-06-10T15:10:52+03:00</created_at>
  <id type="integer">6</id>
  <label>BSZ</label>
  <location_group_id type="integer">1</location_group_id>
  <updated_at type="datetime">2013-07-08T18:45:37+03:00</updated_at>
</backup_server_group>
</backup_server_groups>
```

**Where:**

**label** – backup server zone title

**location_group_id** - ID of a location group the backup server zone is assigned to

**id** – backup server zone ID
52 Logs

OnApp logs all cloud management actions that take place on cloud resources, including virtual servers, disks, data stores, compute resources, templates and networks, as well as alerts and notifications.

Currently the following behavior is implemented in OnApp for VS log item lists:

- **GET** /virtual_machines/:id/logs.xml and **GET** /virtual_machines/:id/logs.json requests return the last 10 transactions.
- **GET** /virtual_machines/:id/logs.xml/page/2 and **GET** /virtual_machines/:id/logs.json/page/2 requests return the next 10 transactions.
- Use the **GET** /virtual_machines/:id/logs.xml/per_page/20 and **GET** /virtual_machines/:id/logs.json/per_page/20 requests to change count of returned transactions.

52.1 Get List of Log Items

To get the list of log items, use the following request:

**GET /logs.xml**
**GET /logs.json**

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```xml
<log_items type="array">
  <log_item>
    <created_at type="datetime">2011-07-25T15:26:44+07:00</created_at>
    <target_id type="integer">22386</target_id>
    <updated_at type="datetime">2011-07-25T15:26:44+07:00</updated_at>
    <id type="integer">22903</id>
    <target_type>Transaction</target_type>
    <status>Complete</status>
    <action>ResizeVSWithoutReboot</action>
    <resource_diff_id type="integer">12</resource_diff_id>
  </log_item>
  ...
  <log_item></log_item>
  ...
</log_items>
```

Where:
created_at – time in the [YYYY][MM][DD][T][hh][mm][ss]Z format
updated_at – time in the [YYYY][MM][DD][T][hh][mm][ss]Z format
id – log item ID
target_id – ID of the transaction (item in the transaction list. See Get List of Transactions for details)
target_type – type of log item (either Transaction or Alert).
action - the action name
status - the action status (Complete, Warn, Pending, or Failed)
resource_diff_id - the ID of the resource difference that contains the changes to the resources that have been made during the transaction. On how to view resource differences refer to Get Resource Difference Details.

Page History
v. 5.3
• added the resource_diff_id parameter

52.2 Get Log Item Details

To view the details of a log item, use the following request:
GET /logs/:id.xml
GET /logs/:id.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<log_item>
  <created_at type="datetime">2011-07-25T15:26:44+07:00</created_at>
  <target_id type="integer">22386</target_id>
  <updated_at type="datetime">2011-07-25T15:26:44+07:00</updated_at>
  <id type="integer">22903</id>
  <target_type>Transaction</target_type>
  <status>running</status>
  <action>HotMigrate</action>
  <resource_diff_ids type="integer">2117</resource_diff_ids>
</log_item>
```

For details refer to Get List of Log Items section.
52.3 Get List of VS Log Items

To view the list of log items for a virtual server, use the following request:

GET /virtual_machines/:id/logs.xml
GET /virtual_machines/:id/logs.json

XML Request Example

```
curl -i -X GET -u 'user:userpass' --url http://onapp.test/virtual_machines/1/logs.xml
```

JSON Request Example

```
curl -i -X GET -u 'user:userpass' --url http://onapp.test/virtual_machines/1/logs.json
```

Where:

- **id** - the ID of the virtual server

XML Output Example

```
<log_items type="array">
    <log_item>
        <action>ResizeVSWithoutReboot</action>
        <created_at type="datetime">2015-03-10T13:34:20+02:00</created_at>
        <id type="integer">1</id>
        <status>Complete</status>
        <target_id type="integer">100347</target_id>
        <target_type>Transaction</target_type>
        <updated_at type="datetime">2015-03-10T13:34:20+02:00</updated_at>
        <resource_diff_id type="integer">12</resource_diff_id>
    </log_item>
    <log_item>...</log_item>
</log_items>
```

Where:

- **action** - the action name
- **created_at** - time in the [YYYY][MM][DD][hh][mm][ss]Z format
- **id** - log item ID
- **status** - the action status (Complete, Warn, Pending, or Failed)
- **target_id** - ID of the transaction (item in the transaction list. See Get List of Transactions for details)
- **target_type** - type of log item: either Transaction or Alert
- **updated_at** - time in the [YYYY][MM][DD][hh][mm][ss]Z format
- **resource_diff_id** - the ID of the resource difference that contains the changes to the resources that have been made during the transaction. On how to view resource differences refer to Get Resource Difference Details.
v. 5.3

- added the `resource_diff_id` parameter

### 52.4 Get VS Log Item Details

To view the VS log item details, use the following request:

GET `/virtual_machines/:id/logs/:log_id.xml`

GET `/virtual_machines/:id/logs/:log_id.json`

**XML Request Example**

```
curl -i -X GET -u 'user:userpass' --url http://onapp.test/virtual_machines/12/logs/127772.xml
```

**JSON Request Example**

```
curl -i -X GET -u 'user:userpass' --url http://onapp.test/virtual_machines/12/logs/127772.json
```

**Where:**

- **id** - the ID of the virtual server
- **log_id** - log item ID

**XML Output Example**

```
<log_item>
  <action>ResizeVSWithoutReboot</action>
  <created_at type="datetime">2015-03-10T13:34:20+02:00</created_at>
  <id type="integer">127772</id>
  <status>Complete</status>
  <target_id type="integer">100347</target_id>
  <target_type>Transaction</target_type>
  <resource_diff_id type="integer">12</resource_diff_id>
  <updated_at type="datetime">2015-03-10T13:34:20+02:00</updated_at>
</log_item>
```

**Where:**

- **action** - the action name
- **created_at** - time in the [YYYY][MM][DD][hh][mm][ss]Z format
- **id** - log item ID
- **status** - the action status (Complete, Warn, Pending, or Failed)
- **target_id** - ID of the transaction (item in the transaction list. See [Get List of Transactions](#) for details)
- **target_type** - type of log item: either Transaction or Alert
- **updated_at** - time in the [YYYY][MM][DD][hh][mm][ss]Z format
- **resource_diff_id** - the ID of the resource difference that contains the changes to the resources that have been made during the transaction. On how to view resource differences refer to [Get Resource Difference Details](#).
52.5 Get List of Resource Differences

To get the list of resource differences, use the following request:

GET /resource_diffs.xml
GET /resource_diffs.json

XML Request Example

```bash
curl -i -X GET -u user:userpass --url http://onapp.test/resource_diffs.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass --url
'Content-type: application/json'
```

XML Output Example
<resource_diffs type="array">
  <resource_diff>
    <id type="integer">3778</id>
    <transaction_id type="integer">37025</transaction_id>
    <diff type="array">
      <cpus>
        <before/>
        <after>1</after>
      </cpus>
      <label>
        <before/>
        <after>EdgeStreaming</after>
      </label>
      <memory>
        <before/>
        <after>2048</after>
      </memory>
      <cpu_units>
        <before/>
        <after>10</after>
      </cpu_units>
      <cpu_shares>
        <before/>
        <after>1</after>
      </cpu_shares>
      <identifier>
        <before/>
        <after>uhdybkmeruiltyj</after>
      </identifier>
    </diff>
    <created_at type="dateTime">2019-05-06T13:56:24+03:00</created_at>
    <updated_at type="dateTime">2019-05-06T13:56:24+03:00</updated_at>
  </resource_diff>
  ...</resource_diffs>

Where:

id - the ID of the resource difference
transaction_id - the ID of the transaction in which resources have undergone changes
diff - the array of resource which have or have not been changed
before - the value of the resource before the transaction
after - the value of the resource after the transaction
approved_at - this parameter is not currently applicable to resource differences
created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
updated_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

52.6 Get Resource Difference Details

To view the details of a resource difference, use the following request:

GET /resource_diffs/:resource_diff_id.xml
GET /resource_diffs/:resource_diff_id.json
resource_diff_id can be retrieved with **Get List of Resource Differences** API request.

**XML Request Example**

```bash
curl -i -X GET http://onapp.test/resource_diffs/2717.xml -u user:userpass
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X GET http://onapp.test/resource_diffs/2717.json -u user:userpass
-H 'Accept: application/json' -H 'Content-type: application/json'
```

**XML Output Example**

```xml
<resource_diff>
  <id type="integer">2717</id>
  <transaction_id type="integer">29939</transaction_id>
  <diff type="array">
    <id type="integer">2717</id>
    <hypervisor_id type="array">
      <before>49</before>
      <after>50</after>
    </hypervisor_id>
    <hypervisor_label type="array">
      <before>KVM C7 HV1 SDN</before>
      <after>KVM C7 HV2 SDN</after>
    </hypervisor_label>
    <hypervisor_ip_address type="array">
      <before>10.0.24.21</before>
      <after>10.0.24.22</after>
    </hypervisor_ip_address>
  </diff>
  <created_at type="datetime">2019-03-01T15:40:58.000+02:00</created_at>
  <updated_at type="datetime">2019-03-01T15:40:58.000+02:00</updated_at>
</resource_diff>
```

The output above is for the Hot/Cold Migrate VS transaction. For other transactions, the output may contain different resources.

**Where:**

- **id** - the ID of the resource difference
- **transaction_id** - the ID of the transaction in which resources have undergone changes
- **diff** - the array of resource which have or have not been changed
- **before** - the value of the resource before the transaction
- **after** - the value of the resource after the transaction
- **hypervisor_id** - the ID of the compute resource
hypervisor_label - the name of the compute resource
hypervisor_ip_address - the IP address of the compute resource
created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
53 Look&Feel

You can edit various aspects of the Control Panel’s look & feel, including the logo displayed, background colors and other graphics.

53.1 Look&Feel Themes

This chapter includes API requests for creating and managing the look of OnApp Control Panel.

53.1.1 Get List of Look&Feel Themes

To get the list of Look&Feel themes, use the following request:

GET /settings/themes.xml
GET /settings/themes.json

XML Request Example


JSON Request Example


XML Output Example
<theme>

<id type="integer">3</id>
<label>test</label>
<active type="boolean">false</active>
<created_at type="dateTime">2016-12-01T15:47:49+02:00</created_at>
<updated_at type="dateTime">2016-12-01T15:48:45+02:00</updated_at>
</theme>

Where:

- **id** - the ID of the theme
- **label** - the name of the theme
- **active** - whether the theme is active or not
- **created_at** – the date when the theme was added
- **updated_at** – the date when the theme was updated
- **application_title** - the desired title which will be displayed at the top left corner of the browser window
- **logo** - a custom logo
- **url** - the route, from where you upload the logo
- **disable_logo** - set 'true' to disable the custom logo
- **favicon** - a custom favicon logo.
- **url** - the route, from where you upload the favicon logo
- **disable_favicon** - set 'true' to disable the custom favicon logo
powered_by_hide - set 'true' to remove the Powered by OnApp message at the top of the navigation pane
powered_by_url - enter an URL you wish to link to instead of http://www.onapp.com/
powered_by_link_title - enter a name for URL you wish to link to
powered_by_color - this is the color displayed in the main body of the page (e.g. behind the fields you're currently editing)
powered_by_text - specify the text which will be added after Powered by instead of OnApp
wrapper_background_color - this is the color displayed around the rest of the UI
wrapper_top_background_image - a custom wrapper top background image
wrapper_bottom_background_image - a custom wrapper bottom background image
disable_wrapper_bottom_background_image - set 'true' to disable wrapper bottom background image
body_background_color - the color of body background
body_background_image - a custom image for body background
url - the route, from where you upload the custom image for body background
disable_body_background_image - set 'true' to prevent the top background image displaying
html_header - enter the html codes to display instead of default header
html_footer - enter the html codes to display instead of default footer
user_group_ids - an array of ID of user groups, for which you wish to apply the theme

53.1.2 Get Look&Feel Theme Details
To get the Look&Feel theme details, use the following request:
GET /settings/themes/:id.xml
GET /settings/themes/:id.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<theme>
  <id type="integer">3</id>
  <label>test</label>
  <active type="boolean">false</active>
  <created_at type="dateTime">2016-12-01T15:47:49+02:00</created_at>
  <updated_at type="dateTime">2016-12-01T15:48:45+02:00</updated_at>
  <application_title/>
  <logo>
    <url>/themes/9ae04c286680147c7bb412f6b1784fe/logo-e5543824bcd844728a03bc134656ac7.png</url>
  </logo>
  <disable_logo type="boolean">false</disable_logo>
  <favicon>
    <url>/themes/9ae04c286680147c7bb412f6b1784fe/favicon-false97a0c003827c52b3e866f8.png</url>
  </favicon>
  <disable_favicon type="boolean">false</disable_favicon>
  <powered_by_hide type="boolean">false</powered_by_hide>
  <powered_by_url/>
  <powered_by_link_title nil="true"/>
  <powered_by_color/>
  <powered_by_text/>
  <wrapper_background_color nil="true"/>
  <wrapper_top_background_image nil="true"/>
  <disable_wrapper_top_background_image nil="true"/>
  <wrapper_bottom_background_image nil="true"/>
  <disable_wrapper_bottom_background_image nil="true"/>
  <body_background_color/>
  <body_background_image>
    <url nil="true"/>
  </body_background_image>
  <disable_body_background_image type="boolean">false</disable_body_background_image>
  <html_header/>
  <html_footer/>
  <user_group_ids type="array"/>
</theme>

Where:

- **id** - the ID of the theme
- **label** - the name of the theme
- **active** - whether the theme is active or not
- **created_at** – the date when the theme was added
- **updated_at** – the date when the theme was updated
- **application_title** - the desired title which will be displayed at the top left corner of the browser window
- **logo** - a custom logo
- **url** - the route, from where you upload the logo
- **disable_logo** - set 'true' to disable the custom logo
- **favicon** - a custom favicon logo.
- **url** - the route, from where you upload the favicon logo
- **disable_favicon** - set 'true' to disable the custom favicon logo
powered_by_hide - set 'true' to remove the Powered by OnApp message at the top of the navigation pane
powered_by_url - enter an URL you wish to link to instead of http://www.onapp.com/
powered_by_link_title - enter a name for URL you wish to link to
powered_by_color - this is the color displayed in the main body of the page (e.g. behind the fields you're currently editing)
powered_by_text - specify the text which will be added after Powered by instead of OnApp
wrapper_background_color - this is the color displayed around the rest of the UI
wrapper_top_background_image - a custom wrapper top background image
wrapper_bottom_background_image - a custom wrapper bottom background image
disable_wrapper_bottom_background_image - set 'true' to disable wrapper bottom background image
body_background_color - the color of body background
body_background_image - a custom image for body background
url - the route, from where you upload the custom image for body background
disable_body_background_image - set 'true' to prevent the top background image displaying
html_header - enter the html codes to display instead of default header
html_footer - enter the html codes to display instead of default footer
user_group_ids - an array of ID of user groups, for which you wish to apply the theme

53.1.3 Add Look&Feel Theme
To add a Look&Feel theme, use the following request:
POST /settings/themes.xml
POST /settings/themes.json

XML Request Example

```bash
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<?xml version="1.0" encoding="UTF-8"?><theme><label>test</label><user_group_ids type="array"><user_group_id>1323</user_group_id></user_group_ids></theme>' --url http://onapp.test/settings/themes.xml
```

JSON Request Example

```bash
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"theme":{"label":"test","user_group_ids":["1","2"]}}' --url http://onapp.test/settings/themes.json
```

Where:

* label* - the name of the theme
* user_group_ids - the user groups, for which you wish to apply the theme

For more parameters, that can be added, refer to the Get Look&Feel Theme Details.
53.1.4 Edit Look&Feel Theme

To edit details of a Look&Feel theme, use the following request:

PUT /settings/themes/:id.xml
PUT /settings/themes/:id.json

**XML Request Example**

```
curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<theme><label>test</label><user_group_ids type="array"><user_group_id>1323</user_group_id></user_group_ids></theme>' --url http://onapp.test/settings/themes/12.xml
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"theme":{"label":"test","user_group_ids":["1","2"]}}' --url http://onapp.test/settings/themes/12.json
```

Where:

- `label` - the name of the theme
- `user_group_ids` - the user groups, for which you wish to apply the theme

For more parameters, that can be edited, refer to the Get Look&Feel Theme Details.

53.1.5 Delete Look&Feel Theme

To delete a Look&Feel theme, use the following request:

DELETE /settings/themes/:id.xml
DELETE /settings/themes/:id.json

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/themes/12.xml
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/themes/12.json
```

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no Look&Feel theme with a requested ID, or URL is incorrect.

53.2 Service Insertion Groups

Service insertion groups are containers for service insertion pages. This section contains the API requests you can use to manage service insertion groups.
53.2.1 Get List of Service Insertion Groups

To get the list of service insertion groups, use the following request:

GET /settings/sif/groups.xml
GET /settings/sif/groups.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<groups type="array">
<group>
<id type="integer">1</id>
<label>test</label>
<weight type="integer">10</weight>
<global type="boolean">true</global>
<created_at type="dateTime">2017-07-11T16:26:14+03:00</created_at>
<updated_at type="dateTime">2017-07-11T16:26:14+03:00</updated_at>
</group>
</groups>
```

Where:

- `id` - the ID of the service insertion group
- `label` - the name of the service insertion group
- `weight` - value from 10 to 0 to determine which group comes first
- `global` - true if group is available to all users; otherwise false
- `created_at` - the date when the service insertion group was added
- `updated_at` - the date when the service insertion group was updated

53.2.2 Get Service Insertion Group Details

To get the service insertion group details, use the following request:

GET /settings/sif/groups/:id.xml
GET /settings/sif/groups/:id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<group>
  <id type="integer">1</id>
  <label>test</label>
  <weight type="integer">10</weight>
  <global type="boolean">true</global>
  <created_at type="dateTime">2017-07-11T16:26:14+03:00</created_at>
  <updated_at type="dateTime">2017-07-11T16:26:14+03:00</updated_at>
</group>
```

Where:

- **id** - the ID of the service insertion group
- **label** - the name of the service insertion group
- **weight** - value from 10 to 0 to determine which group comes first
- **global** - true if group is available to all users; otherwise false
- **created_at** – the date when the service insertion group was added
- **updated_at** – the date when the service insertion group was updated

53.2.3 Add Service Insertion Group

To add a service insertion group, use the following request:

POST /settings/sif/groups.xml

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:

- **label** - the name of the service insertion group
- **weight** - value from 10 to 0 to determine which group comes first
- **global** - true if group is available to all users; otherwise false
53.2.4 Edit Service Insertion Group
To edit details of a service insertion group, use the following request:

PUT /settings/sif/groups/:id.xml
PUT /settings/sif/groups/:id.json

XML Request Example

curl -i -X PUT -u user:password -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
"<iframe_group><label>test</label><weight>10</weight><global>true</global>
</iframe_group>" --url http://onapp.test/settings/sif/groups/1.xml

JSON Request Example

curl -i -X PUT -u user:password -H 'Accept: application/json' -H 'Content-type: application/json' -d
"{"iframe_group":{"label":"test","weight":"10","global":"true"}}" --url http://onapp.test/settings/sif/groups/1.json

Where:

- label: the name of the service insertion group
- weight: value from 10 to 0 to determine which group comes first
- global: true if group is available to all users; otherwise false

53.2.5 Delete Service Insertion Group
To delete a service insertion group, use the following request:

DELETE /settings/sif/groups/:id.xml
DELETE /settings/sif/groups/:id.json

XML Request Example

curl -i -X DELETE -u user:password --url http://onapp.test/settings/sif/groups/1.xml

JSON Request Example

curl -i -X DELETE -u user:password --url http://onapp.test/settings/sif/groups/1.json

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no service insertion group with a requested ID, or URL is incorrect.

53.3 Service Insertion Pages
The Service Insertion Framework allows you to bring other portals into OnApp. Also you can integrate an insertion framework into OnApp which will display a web page within the user
profile in the OnApp Control Panel (legacy mode). This section contains the API requests which you can use to manage service insertion pages.

53.3.1 Get List of Service Insertion Pages

To get the list of service insertion pages, use the following request:

GET /settings/sif/pages.xml
GET /settings/sif/pages.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<iframes type="array">
  <iframe>
    <id type="integer">1</id>
    <url>https://docs.onapp.com/</url>
    <user_credentials nil="true"/>
    <credentials_parameter nil="true"/>
    <custom_field_parameter nil="true"/>
    <custom_name>test</custom_name>
    <created_at type="dateTime">2017-07-14T16:57:10+03:00</created_at>
    <updated_at type="dateTime">2017-07-14T16:57:10+03:00</updated_at>
    <legacy_mode type="boolean">false</legacy_mode>
    <weight type="integer">10</weight>
    <credentials_type/>
  </iframe>
  <iframe>
    <id type="integer">2</id>
    <url>test.com/</url>
    <user_credentials>Login</user_credentials>
    <credentials_parameter>Login</credentials_parameter>
    <custom_fieldParameter>nationality</custom_field>
    <custom_name>Test</custom_name>
    <created_at type="dateTime">2017-07-17T16:10:45+03:00</created_at>
    <updated_at type="dateTime">2017-07-17T16:10:45+03:00</updated_at>
    <legacy_mode type="boolean">false</legacy_mode>
    <weight type="integer">10</weight>
    <credentials_type>user</credentials_type>
  </iframe>
</iframes>
```

Where:

- **id** - the ID of the service insertion page
- **url** - the URL, which will be displayed in the frame
- **user_credentials** - select the required user parameter value
credentials_parameter - fill in the user field label

custom_field - select the required custom field value

custom_field_parameter - fill in the custom field label

custom_name - the name for the service insertion page

created_at – the date when the service insertion page was added

updated_at – the date when the service insertion page was updated

legacy_mode - true if legacy mode is enabled; otherwise false

weight - value from 10 to 0 to determine which page comes first

credentials_type - indicate credentials type (Global static, User based or User group based)

53.3.2 Get Service Insertion Page Details

To get the service insertion page details, use the following request:

GET /settings/sif/pages/:id.xml
GET /settings/sif/pages/:id.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```xml
<iframe>  
  <id type="integer">
    1
  </id>  
  <url>https://docs.onapp.com/</url>  
  <user_credentials nil="true"/>  
  <credentials_parameter nil="true"/>  
  <custom_field nil="true"/>  
  <custom_field_parameter nil="true"/>  
  <custom_name>test</custom_name>  
  <created_at type="dateTime">2017-07-14T16:57:10+03:00</created_at>  
  <updated_at type="dateTime">2017-07-14T16:57:10+03:00</updated_at>  
  <legacy_mode type="boolean">false</legacy_mode>  
  <weight type="integer">10</weight>  
  <credentials_type/>
</iframe>
```

**Where:**

- **id** - the ID of the service insertion page
- **url** - the URL, which will be displayed in the frame
- **user_credentials** - select the required user parameter value
- **credentials_parameter** - fill in the user field label
**custom_field** - select the required custom field value

**custom_field_parameter** - fill in the custom field label

**custom_name** - the name for the service insertion page

**created_at** – the date when the service insertion page was added

**updated_at** – the date when the service insertion page was updated

**legacy_mode** - true if legacy mode is enabled; otherwise false

**weight** - value from 10 to 0 to determine which page comes first

**credentials_type** - indicate credentials type (Global static, User based or User group based)

### 53.3.3 Add Service Insertion Page

To add a service insertion page, use the following request:

**POST /settings/sif/pages.xml**

**POST /settings/sif/pages.json**

**XML Request Example**

```
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<iframe><custom_name>test</custom_name><weight>10</weight><credentials_type>user_based</credentials_type><url>test.com</url><legacy_mode>false</legacy_mode><user_credentials>login</user_credentials><credentials_parameter>login</credentials_parameter><custom_field>nationality</custom_field><custom_field_parameter>nt</custom_field_parameter><iframe_group_ids type="array"><iframe_group_id>6</iframe_group_id></iframe_group_ids></iframe>' --url http://onapp.test/settings/sif/pages.xml
```

**JSON Request Example**

```
```

**Where:**

- **url** - the URL, which will be displayed in the frame
- **user_credentials** - select the required user parameter value
- **credentials_parameter** - fill in the user field label
- **custom_field** - select the required custom field value
- **custom_field_parameter** - fill in the custom field label
- **custom_name** - the name for the service insertion page
- **legacy_mode** - true if legacy mode is enabled; otherwise false
- **weight** - value from 10 to 0 to determine which page comes first
- **credentials_type** - indicate credentials type (Global static, User based or User group based)
iframe_group_ids - an array of service insertion group IDs, to which the service insertion page belongs

53.3.4 Edit Service Insertion Page

To edit details of a service insertion page, use the following request:

PUT /settings/sif/pages/:id.xml
PUT /settings/sif/pages/:id.json

**XML Request Example**

curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
"<iframe><custom_name>test</custom_name><weight>10</weight><credentials_type>user_based</credentials_type><url>test.com</url><legacy_mode>false</legacy_mode><user_credentials>login</user_credentials><credentials_parameter>test</credentials_parameter><custom_field>nationality</custom_field><custom_field_parameter>nt</custom_field_parameter><iframe_group_ids type="array"><iframe_group_id>6</iframe_group_id></iframe_group_ids></iframe>" --url http://onapp.test/settings/sif/pages/12.xml

**JSON Request Example**

curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d
'{"iframe":{"custom_name":"test","weight":"10","credentials_type":"user_based","url":"test.com","legacy_mode":false,"user_credentials":"login","credentials_parameter":"test","custom_field":"nationality","custom_field_parameter":"nt","iframe_group_ids":["6"]}}' --url http://onapp.test/settings/sif/pages/12.json

**Where:**
url - the URL, which will be displayed in the frame
user_credentials - select the required user parameter value
credentials_parameter - fill in the user field label
custom_field - select the required custom field value
custom_field_parameter - fill in the custom field label
custom_name - the name for the service insertion page
legacy_mode - true if legacy mode is enabled; otherwise false
weight - value from 10 to 0 to determine which page comes first
credentials_type - indicate credentials type (Global static, User based or User group based)
iframe_group_ids - an array of service insertion group IDs, to which the service insertion page belongs

53.3.5 Delete Service Insertion Page

To delete a service insertion page, use the following request:

DELETE /settings/sif/pages/:id.xml
DELETE /settings/sif/pages/:id.json

XML Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/sif/pages/12.xml
```

JSON Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/sif/pages/12.json
```

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no service insertion page with a requested ID, or URL is incorrect.
54 My Template Groups

My Template Groups is the class that organizes all custom templates of the user into separate groups. Each template group can be associated with specific licensing type for Windows based templates. The user can only access and manage his own template groups.

54.1 Get My Template Groups List

To view the list of your template groups, use the following request:

GET /image_template_groups/own.xml
GET /image_template_groups/own.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output example**
<image_template_groups type="array">
  <image_template_group>
    <id type="integer">48</id>
    <parent_id nil="true"/>
    <lft type="integer">49</lft>
    <rgt type="integer">52</rgt>
    <mak type="boolean">false</mak>
    <own type="boolean">true</own>
    <kms type="boolean">false</kms>
    <kms_server_label>serverlabel</kms_server_label>
    <kms_host>server.host</kms_host>
    <kms_port>2</kms_port>
    <created_at type="datetime">2014-04-25T11:36:36+00:00</created_at>
    <updated_at type="datetime">2014-04-25T11:36:36+00:00</updated_at>
    <system_group type="boolean">false</system_group>
    <hypervisor_group_id nil="true"/>
    <user_id type="integer">2508</user_id>
    <children type="array">
      <child>
        <id type="integer">49</id>
        <parent_id type="integer">48</parent_id>
        <lft type="integer">50</lft>
        <rgt type="integer">51</rgt>
        <mak type="boolean">false</mak>
        <own type="boolean">true</own>
        <kms type="boolean">true</kms>
        <kms_server_label>serverlabel</kms_server_label>
        <kms_host>server.host</kms_host>
        <kms_port>2</kms_port>
        <created_at type="datetime">2014-04-25T11:43:56+00:00</created_at>
        <updated_at type="datetime">2014-04-25T12:08:26+00:00</updated_at>
        <system_group type="boolean">false</system_group>
        <hypervisor_group_id nil="true"/>
        <user_id type="integer">2508</user_id>
        <children type="array"/>
        <relations type="array">
          <relation>
            <id type="integer">944</id>
            <template_id type="integer">1043</template_id>
            <image_template_group_id type="integer">49</image_template_group_id>
            <price type="decimal">0.0</price>
            <created_at type="datetime">2014-04-25T12:07:43+00:00</created_at>
            <updated_at type="datetime">2014-04-25T12:07:43+00:00</updated_at>
          </relation>
          <relation>
            <id type="integer">1043</id>
            <label>win_custom_template</label>
            <created_at type="datetime">2014-04-15T15:21:15+00:00</created_at>
            <updated_at type="datetime">2014-04-15T15:36:37+00:00</updated_at>
          </relation>
        </relations>
      </child>
    </children>
  </image_template_group>
</image_template_groups>
<min_disk_size type="integer">20</min_disk_size>
<user_id type="integer">2508</user_id>
<template_size type="integer">16131528</template_size>
<allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
<operating_system_arch>x64</operating_system_arch>
<operating_system_edition>STD</operating_system_edition>
<operating_system_tail>R2</operating_system_tail>
<parent_template_id type="integer">36</parent_template_id>
<virtualization>xen</virtualization>
<min_memory_size type="integer">1024</min_memory_size>
<disk_target_device>xen: hd
kvm: hd</disk_target_device>
<cdn type="boolean">false</cdn>
<backup_server_id type="integer">2</backup_server_id>
<smart_server type="boolean">false</smart_server>
<baremetal_server type="boolean">false</baremetal_server>
<initial_password nil="true"/>
<initial_username nil="true"/>
<remote_id nil="true"/>
<manager_id nil="true"/>
</image_template>
</relation>
</child>
</children>
<relations type="array"/>
</image_template_group>
</image_template_groups>

Where:

image_template_group – the array with the template group details
id – the ID of this template group
label – the name of the template group
parent_id – the ID of the parent group. Applicable to child groups only
lft - left nested set identifier
rgt - right nested set identifier
depth - the depth of a given node (distance from this template group to the root)
mak – MAK windows licensing type
own – user's own license for Windows licensing
kms – KMS windows licensing type
kms_server_label – name of the KMS licensing server
kms_host - KMS server host name
kms_port – KMS server port
created_at – time when the template group was created
updated_at – time of the last changes to the template group
system_group - true for the groups created with the OnApp installation; for Template Store groups only
user_id – user, to whom this group belongs
children – an array of the child groups
child – an array with child group details
id – child group ID
label – child group name
parent_id – the ID of the group to which this child group belongs
relations – an array with templates associated with this group/childgroup
id – the relation ID
template_id – the ID of the assigned template
image_template_group_id – the ID of the group/childgroup, to which the above template is
assigned
price – the template’s cost
image_template – an array with template details
id – the template’s ID
label – the name of the template
version – the file’s version
file_name – the name of the template file
operating_system – operating system name
operating_system_distro – operating system distribution
allowed_swap - true, if the swap is allowed, otherwise false
state - state of the template (active, inactive)
checksum – file checksum
allow_resize_without_reboot – true if resize without reboot is possible; otherwise false
min_disk_size – minimum disk size required to build a VS on this template (GB)
user_id - the ID of a user who owns this template
template_size - the size of the template
allowed_hot_migrate - true, if the hot migration is allowed, otherwise false
operating_system_arch – architecture of the operating system
operating_system_edition – edition of the OS
operating_system_tail – tail of the OS
parent_template_id – the ID of the template from which this custom template originates
virtualization – virtualization type which is compatible with this template
min_memory_size – minimal required RAM for the template
disk_target_device - the prefix indicating the method of translating the disk to a VS by compute resource
cdn - true if this template can be used for building edge servers, otherwise false
backup_server_id - the ID of the backup server where the template is stored
ext4 - true if ext4 file system is supported
smart_server - true if the smart server can be built from this template
baremetal_server - true if the baremetal server can be built from this template
initial_password - preset the password for the VS built on this template
initial_username - preset the username for the VS built on this template
remote_id - ID of the template, if it came from the market
manager_id - ID of the template on the template server

54.2 Get My Template Group Details

To get details of a particular template group, use the following request:

GET /settings/image_template_groups/:image_template_group_id.xml
GET /settings/image_template_groups/:image_template_group_id.json

XML Request Example

```bash
curl -i -X GET -u user:userpass
http://onapp.test/settings/image_template_groups/57.xml
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass
http://onapp.test/settings/image_template_groups/57.json
```

XML Output Example
Where:

- **image_template_group** – the array with the template group details
- **id** – the ID of this template group
- **label** – the name of the template group
- **parent_id** – the ID of the parent group. Applicable to child groups only
- **lft** - left nested set identifier
- **rgt** - right nested set identifier
- **depth** - the depth of a given node (distance from this template group to the root)
- **mak** – MAK windows licensing type
- **own** – user’s own license for Windows licensing
- **kms** – KMS windows licensing type
- **kms_server_label** – name of the KMS licensing server
- **kms_host** - KMS server host name
- **kms_port** – KMS server port
- **created_at** – time when the template group was created
- **updated_at** – time of the last changes to the template group
- **system_group** - true for the groups created with the OnApp installation; for Template Store groups only
- **hypervisor_group_id** - compute zone ID
- **user_id** – user, to whom this group belongs
- **children** – an array of the child groups

54.3 Add My Template Group

To add a template group, use the following request:

POST /settings/image_template_groups.xml
POST /settings/image_template_groups.json

XML Request Example
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password -d
'"user_id":2,"label":"zaza","mak":1,"kms_host":"ededde.fe","kms_port":"5453","kms":1,"own":0,"kms_server_label":"wqqsasawqw"'
'--urlhttp://onapp.test/settings/image_template_groups.xml

JSON Request Example

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:password -d
'"image_template_group":{"user_id":2, "label":"zaza", "mak":1,"kms_host":"ededde.fe", "kms_port":"5453", "kms":1, "own":1, "kms_server_label":"wqqw"}'
'--urlhttp://onapp.test/settings/image_template_groups.json
```

Where:

- **label** – the name of the template group
- **mak** – MAK windows licensing type
- **own** – user's own license for Windows licensing
- **kms** – KMS Windows licensing type
- **kms_server_label** – name of the KMS licensing server; required parameter if the KMS licensing type was selected
- **kms_host** - KMS server hostname; required parameter if the KMS licensing type was selected
- **kms_port** – KMS server port; required parameter if the KMS licensing type was selected
- **user_id** – user, to whom this group belongs

Returns 201 HTTP response on success

XML Output Example

```
<image_template_group>
  <created_at type="datetime">2012-07-13T03:55:21-10:00</created_at>
  <depth type="integer">0</depth>
  <id type="integer">106</id>
  <kms type="boolean">true</kms>
  <kms_host>ededde.fe</kms_host>
  <kms_port>5453</kms_port>
  <kms_server_label>wqqsasawqw</kms_server_label>
  <label>zaza</label>
  <lft type="integer">85</lft>
  <mak type="boolean">true</mak>
  <own type="boolean">false</own>
  <parent_id nil="true"/>
  <rgt type="integer">86</rgt>
  <updated_at type="datetime">2012-07-13T03:55:21-10:00</updated_at>
  <system_group type="boolean">false</system_group>
  <hypervisor_group_id nil="true"/>
  <user_id type="integer">2508</user_id>
  <children type="array"/>
  <relations type="array"/>
</image_template_group>
```
54.4 Add Child Template Group

To add a child template group, use the following request:

**POST** http://onapp.test/settings/image_template_groups.xml
**POST** http://onapp.test/settings/image_template_groups.json

**XML Request Example**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password -d '
  <image_template_group>
    <user_id>2</user_id>
    <label>zaza</label>
    <parent_id>50</parent_id>
    <mak>1</mak>
    <kms_host>ededde.fe</kms_host>
    <kms_port>5453</kms_port>
    <kms>1</kms>
    <own>0</own>
    <kms_server_label>enother</kms_server_label>
  </image_template_group>'
--url http://onapp.test/settings/image_template_groups.xml
```

**JSON Request Example**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:password -d '
  {"image_template_group": {"user_id": "2", "label": "zaza_ch1", "parent_id": "51", "mak": "1", "kms_host": "ededde.fe", "kms_port": "5453", "kms": "1", "own": "1", "kms_server_label": "wqwqw"}}'
--url http://onapp.test/settings/image_template_groups.json
```

**Where:**

- **label** *–* the name of the child template group
- **mak** – MAK windows licensing type
- **own** – user's own license for Windows licensing
- **kms** – KMS Windows licensing type
- **kms_server_label** – name of the KMS licensing server; required parameter if the the KMS licensing type was selected
- **kms_host** - KMS server host name; required parameter if the the KMS licensing type was selected
- **kms_port** – KMS server port; required parameter if the the KMS licensing type was selected
- **user_id** *–* user, to whom this group belongs
- **parent_id** *–* the ID of the parent template group

To edit or delete a child group, use the same requests as for template groups.

54.5 Edit My Template Group

To edit details of a template group, use the following request:

**PUT** /settings/image_template_groups/:id.xml
**PUT** /settings/image_template_groups/:id.json

**XML Request Example**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password -d '
  <image_template_group>
    <user_id>2</user_id>
    <label>zaza</label>
    <parent_id>50</parent_id>
    <mak>1</mak>
    <kms_host>ededde.fe</kms_host>
    <kms_port>5453</kms_port>
    <kms>1</kms>
    <own>0</own>
    <kms_server_label>enother</kms_server_label>
  </image_template_group>'
--url http://onapp.test/settings/image_template_groups.xml
```
54.6 Delete My Template Group

To delete a template group, use the following request:

DELETE /settings/image_template_groups/:id.xml
DELETE /settings/image_template_groups/:id.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/image_template_groups/12.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/image_template_groups/12.xml
```

54.7 Get List of Templates Attached to Template Group

To see the list of attached templates, use the following request:

GET /settings/image_template_groups/:image_template_group_id/relation_group_templates.xml
GET
/settings/image_template_groups/:image_template_group_id/relation_group_templates.json

XML Request Example


JSON Request Example


XML Output Example
<relation_group_templates type="array">
  <relation_group_template>
    <created_at type="datetime">2014-04-25T12:07:43+00:00</created_at>
    <id type="integer">944</id>
    <image_template_group_id type="integer">49</image_template_group_id>
    <price type="decimal">0.0</price>
    <template_id type="integer">1043</template_id>
    <updated_at type="datetime">2014-04-25T12:07:43+00:00</updated_at>
    <image_template>
      <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
      <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
      <allowed_swap type="boolean">false</allowed_swap>
      <backup_server_id type="integer">2</backup_server_id>
      <baremetal_server type="boolean">false</baremetal_server>
      <cdn type="boolean">false</cdn>
      <checksum>1f6e8532a3b914be5d02e5ed178396e1</checksum>
      <created_at type="datetime">2014-04-15T15:21:15+00:00</created_at>
      <disk_target_device>xen: hd
            kvm: hd</disk_target_device>
      <ext4 type="boolean">false</ext4>
      <file_name>m3xp52x2yog50g_20140415182115</file_name>
      <id type="integer">1043</id>
      <initial_password>Password1</initial_password>
      <initial_username>Administrator</initial_username>
      <label>win_custom_template</label>
      <manager_id nil="true"/>
      <min_disk_size type="integer">20</min_disk_size>
      <min_memory_size type="integer">1024</min_memory_size>
      <operating_system>windows</operating_system>
      <operating_system_arch>x64</operating_system_arch>
      <operating_system_distro>2008</operating_system_distro>
      <operating_system_edition>STD</operating_system_edition>
      <operating_system_tail>R2</operating_system_tail>
      <parent_template_id type="integer">36</parent_template_id>
      <remote_id nil="true"/>
      <smart_server type="boolean">false</smart_server>
      <state>active</state>
      <template_size type="integer">16131528</template_size>
      <updated_at type="datetime">2014-04-15T15:36:37+00:00</updated_at>
      <user_id type="integer">2508</user_id>
      <version>3.2</version>
      <virtualization>xen</virtualization>
    </image_template>
  </relation_group_template>
</relation_group_templates>

Where:

relations – the array of templates associated with this group/childgroup
id – the relation ID
template_id – the ID of the assigned template
image_template_group_id – the ID of the group/childgroup, to which the above template is assigned
price – the template’s cost
image_template – the array with template details
id – the template’s ID
label – the name of the template
version – the file’s version
file_name – the name of the template file
operating_system – operating system name
operating_system_distro – operating system distribution
**allowed_swap** - true, if the swap is allowed, otherwise false

**state** - state of the template (active, inactive)

**checksum** – file checksum

**allow_resize_without_reboot** – true if resize without reboot is possible; otherwise false

**min_disk_size** – minimum disk size required to build a VS on this template (GB)

**user_id** - the ID of a user who owns this template

**template_size** - the size of the template

**allowed_hot_migrate** - true, if the hot migration is allowed, otherwise false

**operating_system_arch** – architecture of the operating system

**operating_system_edition** – edition of the OS

**operating_system_tail** – tail of the OS

**parent_template_id** – the ID of the template from which this custom template originates

**virtualization** – virtualization type which is compatible with this template

**min_memory_size** – minimal required RAM for the template

**disk_target_device** - the prefix indicating the method of translating the disk to a VS by compute resource

**cdn** - true if this template can be used for building edge servers, otherwise false

**backup_server_id** - the ID of the backup server where the template is stored

**ext4** - true if ext4 file system is supported

**smart_server** - true if the smart server can be built from this template

**baremetal_server** - true if the baremetal server can be built from this template

**initial_password** - preset the password for the VS built on this template

**initial_username** - preset the username for the VS built on this template

**remote_id** - ID of the template, if it came from the market

**manager_id** - ID of the template on the template server

### 54.8 Attach Template to Template Group

To attach a template to a group, use the following request:

**POST**

/settings/image_template_groups/:image_template_group_id/relation_group_templates.xml

**POST**

/settings/image_template_groups/:image_template_group_id/relation_group_templates.json

**XML Request Example**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
    '<relation_group_template><template_id>26</template_id></relation_group_template>' --url
    http://onapp.test/settings/image_template_groups/105/relation_group_templates.xml
```

### JSON Request Example

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
    '
    {"relation_group_template":{"template_id":"26"}}
    ' --url
    http://onapp.test/settings/image_template_groups/105/relation_group_templates.json
```
Where:

- `template_id` - is the ID of the template you want to attach

Returns HTTP 201 response on success.

### 54.9 Detach Template from Template Group

To detach a template attached to a template group, use the following request:

```bash
delete /settings/image_template_groups/:image_template_group_id/relation_group_templates/:id.xml
delete /settings/image_template_groups/:image_template_group_id/relation_group_templates/:id.json
```

**XML Request Example**

```
curl -i -X DELETE -u user:userpass http://onapp.test/settings/image_template_groups/12/relation_group_templates/1.xml
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass http://onapp.test/settings/image_template_groups/12/relation_group_templates/1.json
```
55 Network Interfaces

This class represents the methods required to manage network interfaces. Network interfaces connect virtual servers with the network. You can allocate several network interfaces to a VS.

55.1 Get List of VS Network Interfaces

To get the list of network interfaces allocated to this particular VS, use the following request:

GET /virtual_machines/:virtual_machine_id/network_interfaces.xml
GET /virtual_machines/:virtual_machine_id/network_interfaces.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<network_interfaces type="array">
  <network_interface>
    <label>eth0</label>
    <usage nil="true"></usage>
    <created_at type="datetime">2011-03-18T17:45:07+07:00</created_at>
    <updated_at type="datetime">2011-04-08T18:57:20+07:00</updated_at>
    <primary type="boolean">true</primary>
    <usage_month_rolled_at nil="true"></usage_month_rolled_at>
    <id type="integer">502</id>
    <mac_address>00:16:3e:50:35:52</mac_address>
    <usage_last_reset_at nil="true"></usage_last_reset_at>
    <default_firewall_rule>DROP</default_firewall_rule>
    <rate_limit type="integer">0</rate_limit>
    <virtual_machine_id type="integer">518</virtual_machine_id>
    <network_join_id type="integer">4</network_join_id>
    <identifier>pdfjrtpday9el</identifier>
  </network_interface>
  ...
</network_interfaces>
```

Where:

- **label** - network interface name
- **created_at** - the timestamp in the database when this network interface was created
- **updated_at** - the timestamp in the database when this network interface was updated
- **primary** - true if this network interface is primary, otherwise false
id - the ID of this network interface
mac_address – network interface MAC address
rate_limit - port speed in Mbps
identifier - the identifier in the database of this network interface
network_join_id - the ID of the network join to which this network interface belongs
virtual_machine_id - the ID of a virtual server to which this network interface is attached

55.2 Get Network Interface Details

To get a particular network interface details, use the following request:

GET /virtual_machines/:virtual_machine_id/network_interfaces/:id.xml
GET /virtual_machines/:virtual_machine_id/network_interfaces/:id.json

This request will output details for a network interface. The explanation of the fields is the same as for Get List of VS Network Interfaces method.

55.3 Add Network Interface to VS

To add a new network interface, use the following request:

POST /virtual_machines/:virtual_machine_id/network_interfaces.xml
POST /virtual_machines/:virtual_machine_id/network_interfaces.json

XML Request Example

```
```

JSON Request Example

```
```

Where:

- label* - give the label of a network interface you wish to attach
- rate_limit* - set the port speed of a network interface you wish to attach
- network_join_id* - set the ID of a physical network used to attach this network interface
- primary* - set 1 if the interface is primary. Otherwise false.
After adding the network interface, the virtual server should be power cycled for the change to take effect.

### 55.4 Edit Network Interface

To edit network interface details, use the following request:

```
PUT /virtual_machines/:virtual_machine_id/network_interfaces/:id.xml
PUT /virtual_machines/:virtual_machine_id/network_interfaces/:id.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

**Where:**
- `label` - the label of a network interface
- `rate_limit` - the port speed of a network interface

### 55.5 Delete Network Interface

To delete a network interface from a virtual server, use the following request:

```
DELETE /virtual_machines/:virtual_machine_id/network_interfaces/:id.xml
DELETE /virtual_machines/:virtual_machine_id/network_interfaces/:id.json
```

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/virtual_machines/23/networkInterfaces/12.xml
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/virtual_machines/23/networkInterfaces/12.json
```
This returns an HTTP 204 response if the network interface is deleted, or HTTP 404 if the network interface with the specified ID isn't found or the requested URL is incorrect.

55.6 Get VS Network Interface Usage Statistics

To view the network interface usage statistics of a virtual server, use the following request:

GET /virtual_machines/:virtual_machine_id/network_interfaces/:network_interface_id/usage.xml
GET /virtual_machines/:virtual_machine_id/network_interfaces/:network_interface_id/usage.json

Define a shorter period by setting Start and End time in the API call:


XML Output Example

<net_hourly_stats type="array">
  <net_hourly_stat>
    <created_at type="datetime">2016-01-13T14:00:21Z</created_at>
    <data_received type="integer">156</data_received>
    <data_sent type="integer">25</data_sent>
    <id type="integer">14510</id>
    <network_interface_id type="integer">952</network_interface_id>
    <stat_time type="datetime">2016-01-13T14:00:00Z</stat_time>
    <updated_at type="datetime">2016-01-13T14:00:21Z</updated_at>
    <user_id type="integer">7</user_id>
    <virtual_machine_id type="integer">690</virtual_machine_id>
  </net_hourly_stat>
<net_hourly_stat>...
</net_hourly_stats>

Where:

created_at - the timestamp in DB when the record was created

data_received - the number of Kilobytes (KB) received by this VS

data_sent - the number of Kilobytes (KB) sent by this VS

id - the ID of the statistics

network_interface_id - the ID of the network interface
stat_time - the time when statistics were generated
updated_at - the timestamp in DB when the record was updated
user_id - the ID of the VS owner
virtual_machine_id - the ID of the VS
56 Networks

The class enables you to modify network configurations. The network resources available to the entire cloud can be configured in the system Control Panel or via API. Specific network resources can be set up manually, and automatically on VS creation.

56.1 Get List of Networks

To get the list of all networks in the cloud, use the following request:

GET /settings/networks.xml
GET /settings/networks.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<networks type="array">
  <network>
    <label>Public Network</label>
    <created_at type="datetime">2011-02-11T12:46:09+02:00</created_at>
    <network_group_id type="integer">3</network_group_id>
    <updated_at type="datetime">2011-02-11T13:20:09+02:00</updated_at>
    <id type="integer">1</id>
    <vlan type="integer" nil="true"/>
    <identifier>41kgi2ges03kma</identifier>
  </network>
</networks>
```

Where:

- `created_at` - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- `id` - the network ID
- `label` - the optional network label
- `updated_at` - the date when the network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- `vlan` - the VLAN this network belongs to
- `network_group_id` - the ID of the network zone to which this network is attached
- `identifier` - the identifier of the network in the database
56.2 Get Network Details

To get the details of a particular network, use the following request:

GET /settings/networks/:id.xml
GET /settings/networks/:id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<network>
  <label>public</label>
  <created_at type="datetime">2010-10-28T19:55:40+07:00</created_at>
  <updated_at type="datetime">2010-12-29T22:31:15+07:00</updated_at>
  <network_group_id type="integer">2</network_group_id>
  <vlan type="integer">391</vlan>
  <id type="integer">1</id>
  <identifier>hc9fut4iogxt7p</identifier>
</network>
```

Where:

- **label** - the optional Network label
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** - the date when the record was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **network_group_id** - the ID of the network zone to which this network is attached
- **id** - the network ID
- **vlan** - the VLAN this network belongs to
- **identifier** - network identifier

56.3 Add Network

To add a new network, use the following request:

POST /settings/networks.xml
POST /settings/networks.json

**XML Request Example**

```
```

**JSON Request Example**


**Where:**

- `label` - the network name
- `vlan` - the number of a VLAN this network belongs to
- `network_group_id` - the ID of the network zone to which this network is attached
- `type` - the type of the network, set `Networking::Network` for this parameter. This API request creates a shared network - the default type of network in OnApp where a user receives an IP address on the network they have access to.

**Page History**

v. 5.4
- added `type` parameter

### 56.4 Edit Network

To edit a network, use the following request:

PUT /settings/networks/:id.xml
PUT /settings/networks/:id.json

**XML Request Example**


**JSON Request Example**


**Where:**

- `id` – the network ID
label - the network name

vlan - the number of a VLAN this network belongs to

network_group_id - the ID of the network zone to which this network is attached

56.5 Rebuild VS Network

To rebuild the network for a particular VS, use the following request:

POST /virtual_machines/:virtual_machine_id/rebuild_network.xml
POST /virtual_machines/:virtual_machine_id/rebuild_network.json

XML Request Example

curl -X POST -u user:userpass

JSON Request Example

curl -X POST -u user:userpass

Where:

virtual_machine_id - ID of the virtual server

shutdown_type - type of the VS shutdown: hard, graceful or soft

required_startup - set 1 to start up the VS automatically after build, otherwise set 0

force* - set 1 to specify the VS shutdown type ("shutdown_type") and whether VS should start up automatically after the reboot ("required_startup").

PLEASE NOTE: If the force reboot option is enabled, the system tries to enter the virtual server. Then, if the transaction fails, the virtual server will be rebooted.

In case the force reboot option is disabled and system can not enter the virtual server, the network rebuild operation will fail.

56.6 Delete Network

To delete a network, use the following request:

DELETE /settings/networks/:id.xml
DELETE /settings/networks/:id.json
XML Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/networks/12.xml
```

JSON Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/networks/12.json
```

Where you have to specify ID of a network you want to delete.
57 Network Zones

A network zone consists of several networks sharing the same permissions and assigned to the same bucket. Network zones can be attached to compute zones, enabling you to create different tiers of service within your cloud. All API calls are available to this class.

57.1 Get List of Network Zones

To get the list of all the network zones available in your cloud, use the following request:

GET /settings/network_zones.xml
GET /settings/network_zones.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<network-groups type="array">
  <network-group>
    <label>net_p</label>
    <location_group_id type="integer">1</location_group_id>
    <preconfigured_only type="boolean">true</preconfigured_only>
    <provider_vdc_id type="integer">51</provider_vdc_id>
    <created_at type="datetime">2011-01-06T18:45Z</created_at>
    <updated_at type="datetime">2011-01-06T18:45Z</updated_at>
    <id type="integer">3</id>
  </network-group>
</network-groups>
```

Where:

- **label** - the network zone title
- **location_group_id** - ID of a location group the network zone is assigned to
- **preconfigured_only** - whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.
- **provider_vdc_id** - the provider resource pool ID
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at - the date when the Network zone was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

id - the network zone ID

Page History
v. 5.6
- added the provider_vdc_id parameter

v. 4.2
- added the preconfigured_only parameter

v. 3.1
- added the location_group_id parameter

57.2 Get Network Zone Details

To get a particular network zone details, use the following request:

GET /settings/network_zones/:id.xml
GET /settings/network_zones/:id.json

XML Request Example


JSON Request Example


XML Output Example

<network-group>
<label>network_zone</label>
<location_group_id type="integer">1</location_group_id>
<preconfigured_only type="boolean">true</preconfigured_only>
<provider_vdc_id type="integer">51</provider_vdc_id>
<created_at type="datetime">2011-01-06T11:18:45Z</created_at>
<updated_at type="datetime">2011-01-06T11:18:45Z</updated_at>
,id type="integer">8</id>
</network-group>

Where:

label - the network zone title

location_group_id - ID of a location group the network zone is assigned to

preconfigured_only - whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.
**provider_vdc_id** - the provider resource pool ID

**created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**updated_at** - the date when the Network zone was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**id** - the network zone ID

### Page History

- **v. 5.6**: added the **provider_vdc_id** parameter

- **v. 4.2**: added the **preconfigured_only** parameter

- **v. 3.1**: added the **location_group_id** parameter

### 57.3 Add Network Zone

To add a new network zone, use the following request:

**POST /settings/network_zones.xml**

**POST /settings/network_zones.json**

#### XML Request Example

```bash
curl -i -X POST http://onapp.test/settings/network_zones.xml -d
  '<network_group><label>TEST.XML</label><location_group_id>1</location_group_id><preconfigured_only>true</preconfigured_only></network_group>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

#### JSON Request Example

```bash
curl -i -X POST http://onapp.test/settings/network_zones.json -d
  '{"network_group":{"label":"TEST.JSON","location_group_id":1,"preconfigured_only":true}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

- **label** - is new data_store_zone title

- **location_group_id** - ID of a location group you wish to assign the network zone to

- **server_type** - specify the type of this network zone, it can be one of the following: virtual, smart, baremetal or vpc. The vpc type indicates a vCloud Director network zone.

- **preconfigured_only** - specify whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.

### Page History

- **v. 5.3**: added the **server_type** parameter
v. 4.2:
- added the `preconfigured_only` parameter

v. 3.1:
- added the `location_group_id` parameter

### 57.4 Edit Network Zone

To edit a label and an ID of a particular network zone, use the following request:

```plaintext
PUT /settings/network_zones/:id.xml
PUT /settings/network_zones/:id.json
```

**XML Request Example**

```
curl -i -X PUT http://onapp.test/settings/network_zones/12.xml -d
'(<network_group><label>Network_Name</label><location_group_id>1</location_group_id><preconfigured_only>true</preconfigured_only></network_group>') -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X PUT http://onapp.test/settings/network_zones/12.json -d
'{"network_group":
{"label":"Network_name","location_group_id":"1","preconfigured_only":"true"}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

- `label` - is new data_store_zone title
- `location_group_id` - ID of a location group you wish to assign the network zone to. You can change the already assigned location only if there are no network joins, IP addresses or name servers within networks in this zone.
- `preconfigured_only` - specify whether the zone can be used for creating Instance Package VSs only. When this option is enabled, the zone cannot be used during custom virtual server (VSs built by setting resources manually) creation.

### Page History

- **v. 4.2:**
  - added `preconfigured_only` parameter
- **v. 3.1:**
  - added the `location_group_id` parameter

### 57.5 Delete Network Zone

To delete a network zone, use the following request:

```plaintext
DELETE /settings/network_zones/:id.xml
DELETE /settings/network_zones/:id.json
```
XML Request Example


JSON Request Example


You will get a 204 status response on success, and 404 if there is no such a network zone with a requested ID or you entered incorrect URL.

57.6 Attach Network to Network Zone

To attach a network to a network zone, use the following request:

POST /settings/network_zones/:network_zone_id/networks/:id/attach.xml
POST /settings/network_zones/:network_zone_id/networks/:id/attach.json

XML Request Example


JSON Request Example


This request attaches network (:network_id *) to a network zone (:network_zone_id *)

- Network zone must contain only one network type - either usual networks, or VMware customer networks.
- When you add a network to a network zone, it inherits the zone’s type. For more information refer to Zone Types.

57.7 Remove Network from Network Zone

To remove a network from a network zone, use the following request:
POST /settings/network_zones/:network_zone_id/networks/:id/detach.xml
POST /settings/network_zones/:network_zone_id/networks/:id/detach.json

**XML Request Example**
```
curl -X POST
http://onapp.test/settings/network_zones/12/networks/1/detach.xml -u
user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**
```
curl -X POST
http://onapp.test/settings/network_zones/12/networks/1/detach.json -u
user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```
58 Notifications

The Control Panel's Notification menu allows users to configure the notifications for their CP. Notifications setup allows you to select the events about which to notify your users. This chapter contains all API calls for Notification center.

58.1 Event Types

OnApp currently offers two event types, system event types and custom event types. System events are fixed and registered in the system with their IDs and names. Custom events are events that you configure and which can later be selected when setting up a subscription.

58.1.1 Get List of Events

To get the list of events, use the following request:

GET /messaging/events.xml
GET /messaging/events.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```xml
<messaging_messages>
  <messaging_message>
    <id>19</id>
    <created_at>2018-03-14T12:07.000+02:00</created_at>
    <updated_at>2018-03-14T12:07.000+02:00</updated_at>
    <data>
      <message>TestCustomEvents</message>
    </data>
    <topic_name>custom_event_custom_event_TestCustomEvents</topic_name>
  </messaging_message>
  ...
</messaging_messages>
```

**Where:**

- `id` - the ID of the event
- `created_at` - the date in the `[YYYY][MM][DD][hh][mm][ss]Z` format
- `updated_at` - the date when the event was updated in the `[YYYY][MM][DD][hh][mm][ss]Z` format
- `message` - the text of the event
topic_name - the name of the event

58.1.2 Get List of Event Types
To view the list of the system event types, use the following request:
GET /messaging/event_types.xml
GET /messaging/event_types.json

XML Request Example
```
curl -i -X GET -u user:userpass --url
http://onapp.test/messaging/event_types.xml -H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

JSON Request Example
```
curl -i -X GET -u user:userpass --url
http://onapp.test/messaging/event_types.json -H 'Accept: application/json'
-H 'Content-type: application/json'
```

XML Output Example
```
<messaging_topics type="array">
  <messaging_topic>
    <id>1</id>
    <name>all_compute_resources_status</name>
    <created_at>2018-03-05T14:26:00.000+02:00</created_at>
    <updated_at>2018-03-05T14:26:00.000+02:00</updated_at>
    <kind>system</kind>
    <description></description>
    <placeholders type="array">
      <placeholder>message</placeholder>
      <placeholder>name</placeholder>
    </placeholders>
  </messaging_topic>
</messaging_topics>
```

Where:
- id - the ID of the event type
- name - the name of the event type

Currently, the following system event type names exist in OnApp:
- All compute resources status - all compute resources in a compute zone have changed their statuses to Online/Offline/Inactive
- Auto healing failed diagnostics - the disk automatic repair failed due to some errors detected
- Auto healing processing disk repair - the disk automatic repair has been initiated
- Can’t schedule transaction - a transaction could not be scheduled in the cloud
- Autobackup failed - the backup creation limit has been reached
- Daemon notification - the status of the OnApp engine has changed to Active/Up/Inactive
Daily storage health report - the daily storage health report will be sent in the notification
Failed task - a task failed in the cloud
Failover compute resource - deprecated option, it will be removed in the next version
Failover process - failover process has been initiated
Federation new announcement - new notification that will be sent to all the buyers who are subscribed to the selected zone
Federation templates changed - the templates have underwent some changes. These changes may include adding a new template, changing the limits of the existing template or deleting one
Generate hourly stats failed - hourly statistics failed to be generated
Hourly storage health report - the hourly storage health report will be sent in the notification
Compute resource missing CPU flags - a compute resource without CPU flags has been detected in the cloud
Compute resource status - one of the compute resources in the cloud has changed its status to Online/Offline/Inactive
Compute resource group responsive - an unresponsive compute zone has been detected in the cloud
Maintenance mode - the Control Panel has been switched to maintenance mode
Reclaim baremetal compute resource - a baremetal server has been deleted. It has been removed from the DB, but it may remain working. To fully remove the baremetal server it might be required to reboot the compute resource on which it was running.
Service addon event - an event with custom message, which is used during creation of service add-on 'Raise event' action
System resources - a hardware resource of the CP server is exhausted
Processes status - deprecated option, it will be removed in the next version
Wrong activated logical volumes - the system has detected VSs' disks that are either activated on the wrong compute resource or on two compute resources simultaneously
Custom event - this is your custom event configured at Control Panel > Notifications > Event Types > Custom Event types tab
Internal server error - an internal server error occurred in the system
Pending approval - a transaction that requires approval has been requested
Approved - a transaction that requires approval has been approved
Declined - a transaction that requires approval has been declined
created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at - the date when the event was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
description - the description of the event type
kind - the type of the event (e.g. system or custom)
placeholders - for events these are usually message - the text of the event and name - the name of the user who will receive the notification

58.1.3 Add Custom Event Type
To add custom event types, use the following request:
POST /messaging/event_types.xml
POST /messaging/event_types.json

**XML Request Example**


**JSON Request Example**


*Where:*

- **name** - the name of the event
- **description** - your description of the custom event

58.1.4 Edit Custom Event Type

To edit custom event type, use the following request:

PUT /messaging/event_types/:id.xml
PUT /messaging/event_types/:id.json

**XML Request example**

curl -i -X PUT -u user:userpass http://onapp.test/messaging/event_types/43.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<messaging_topic><name><TestCustomEvents_modified></name><description><TestCustomEvents></description></messaging_topic>'

**JSON Request example**


*Where:*

- **name** - the name of the event
- **description** - your description of the custom event
58.1.5 Trigger Custom Event Type

To trigger the custom event, use the following request:

POST /messaging/event_types/:id/custom_event_triggers.xml
POST /messaging/event_types/:id/custom_event_triggers.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**Where:**

- `message` - the triggered message

58.1.6 Delete Custom Event Type

To delete custom events, use the following request:

DELETE /messaging/event_types/:id.xml
DELETE /messaging/event_types/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

58.2 External Recipients

External recipients are users that do not belong to OnApp to whom you want to address certain notifications. External recipients are included in recipients lists beside OnApp users.

58.2.1 Get List of External Recipients

To get the list of external users, use the following request:
GET /messaging/external_users.xml
GET /messaging/external_users.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<messaging_external_users type="array">
  <messaging_external_user>
    <id type="integer">1</id>
    <name>dasfdgd</name>
    <email>aa@sfsd.com</email>
    <created_at type="dateTime">2018-04-11T08:35:43+00:00</created_at>
    <updated_at type="dateTime">2018-04-11T08:35:43+00:00</updated_at>
  </messaging_external_user>
  <messaging_external_user>...</messaging_external_user>
</messaging_external_users>
```

**Where:**

- **id** - the ID of the recipient
- **name** - the name of the recipient
- **email** - the email of the recipient
- **created_at** - the date in the [YYYY][MM][DD]T[hh][mm][ss] format
- **updated_at** - the date when the record was updated in the [YYYY][MM][DD]T[hh][mm][ss] format

### 58.2.2 Add External Recipient

To add external recipients, use the following request:

POST /messaging/external_users.xml
POST /messaging/external_users.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/messaging/external_users.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<messaging_external_user><name>uhjfhngq</name><email>zeigyqwqexternaluser@test.com</email></messaging_external_user>'
```

**JSON Request Example**
curl -i -X POST -u user:userpass --url http://onapp.test/messaging/external_users.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"messaging_external_user": {"name": "uhjfhngq", "email": "zeigyqwqexternaluser@test.com"}}'

Where:

id - the ID of the recipient
name - the name of the recipient
email - the email of the recipient

58.2.3 Edit External Recipient

To edit external recipient, use the following request:

PUT /messaging/external_users.xml
PUT /messaging/external_users.json

XML Request Example

curl -i -X PUT -u user:userpass --url http://onapp.test/messaging/external_users.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<messaging_external_user><name>uhjfhngq</name><email>zeigyqwqexternaluser@test.com</email></messaging_external_user>'

JSON Request Example

curl -i -X PUT -u user:userpass --url http://onapp.test/messaging/external_users.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"messaging_external_user": {"name": "uhjfhngq", "email": "zeigyqwqexternaluser@test.com"}}'

Where:

name - the name of the recipient
email - the email of the recipient

58.2.4 Delete External Recipients

To delete external recipients, use the following request:

DELETE /messaging/external_users/:id.xml
DELETE /messaging/external_users/:id.json

XML Request Example

58.3 Recipients List

Recipients lists determine to whom of your users notifications will be sent. If you want to send notifications to emails that are not registered in your OnApp cloud, you need to add such contacts as external recipients. Recipients lists can include both OnApp users and external recipients.

58.3.1 Get List of Recipients Lists

To get the list of recipients lists, use the following request:

GET /messaging/recipients_lists.xml
GET /messaging/recipients_lists.json

XML Request Example

curl -i -X GET -u 'user:userpass' --url

JSON Request Example

curl -i -X GET -u 'user:userpass' --url

XML Output Example

```xml
<messaging_recipients_lists type="array">
  <messaging_recipients_list>
    <id type="integer">1</id>
    <name>qaav</name>
    <created_at type="dateTime">2017-01-11T14:37:40+00:00</created_at>
    <updated_at type="dateTime">2017-01-11T14:37:40+00:00</updated_at>
  </messaging_recipients_list>
  ...
</messaging_recipients_lists>
```

Where:

- `id` - the ID of the recipients list
- `name` - the name of the recipients list
- `created_at` - the date when the recipients list was created in the [YYYY][MM][DD][hh][mm][ss] format
- `updated_at` - time when the recipients list was updated, in [YYYY][MM][DD][hh][mm][ss] format
58.3.2 Add Recipients List

To add the recipients list, use the following request:

POST /messaging/recipients_lists.xml

POST /messaging/recipients_lists.json

XML Request Example

curl -i -X POST -u user:userpass --url

'default_recipients_list':{
  'name': 'ekujonni',
  'recipient_ids': ["Messaging::ExternalUser21", "User41"]
}'

JSON Request Example

curl -i -X POST -u user:userpass --url

'{"default_recipients_list": {
  "name": "ekujonni",
  "recipient_ids": ["Messaging::ExternalUser21", "User41"]
}}'

Where:

- name - the name of the recipient
- recipient_ids - the array of IDs of recipients

58.3.3 Edit Recipients List

To edit recipients list, use the following request:

PUT /messaging/recipients_lists/:id.xml

PUT /messaging/recipients_lists/:id.json

XML Request Example

curl -i -X PUT -u user:userpass --url

'default_recipients_list':{
  'name': 'updated',
  'recipient_ids': ["Messaging::ExternalUser21", "User41", "Messaging::ExternalUser22"]
}'

JSON Request Example

curl -i -X PUT -u user:userpass --url

'{"default_recipients_list": {
  "name": "updated",
  "recipient_ids": ["Messaging::ExternalUser21", "User41", "Messaging::ExternalUser22"]
}}'
Where:
name - the label of the list
recipient_ids - the array of IDs of recipients

58.3.4 Delete Recipients List
To delete recipient list, use the following request:
DELETE /messaging/recipients_lists/:id.xml
DELETE /messaging/recipients_lists/:id.json

XML Request Example
```
curl -i -X DELETE -u user:userpass --url
```

JSON Request Example
```
curl -i -X DELETE -u user:userpass --url
```

58.4 Notification Templates
Notification templates are message texts that will be sent to your users via email or internal notifications in CP. Notification templates are used when setting up a subscription for your users. There are two types of notification templates: system templates that come pre-installed with OnApp and cannot be deleted but only edited, and custom templates which you add to your cloud.

58.4.1 Get List of Notification Templates
To get the list of notification templates, use the following request:
GET /messaging/notification_templates.xml
GET /messaging/notification_templates.json

XML Request Example
```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/messaging/notification_templates.xml
```

JSON Request Example
```
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/messaging/notification_templates.json
```
XML Output Example

```
<messaging_notification_templates>
  <messaging_notification_template>
    <id type="integer">1</id>
    <name>qa</name>
    <template>%{name} and %{message}</template>
    <created_at type="dateTime">2017-01-11T14:38:02+00:00</created_at>
    <updated_at type="dateTime">2017-01-11T14:38:02+00:00</updated_at>
    <system type="boolean">false</system>
  </messaging_notification_template>
  <messaging_notification_template>...
</messaging_notification_templates>
```

Where:

- **id** - the ID of the notification template
- **name** - the name of the notification template
- **template** - the text of the template
- **created_at** - the date when the template was created in the [YYYY][MM][DD][hh][mm][ss] format
- **updated_at** - the date when the template was updated in the [YYYY][MM][DD][hh][mm][ss] format

58.4.2 Add Notification Template

To add notification templates, use the following request:

**POST /messaging/notification_templates.xml**

**POST /messaging/notification_templates.json**

XML Request Example

```
curl -i -X POST -u user:password --url
http://onapp.test/messaging/notification_templates.xml
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
-d '<messaging_notification_template><name>ewpaczin</name><template>HI %{name}! I'm working %{message}</template></messaging_notification_template>'
```

JSON Request Example

```
curl -i -X POST -u user:password --url
http://onapp.test/messaging/notification_templates.json
-H 'Accept: application/json'
-H 'Content-type: application/json'
-d '{"messaging_notification_template": {"name": "ewpaczin", "template": "HI %{name}! I'm working %{message}"}}'
```

Where:

- **name** - the name of the template
- **template** - the text of the template
58.4.3 Edit Notification Template
To edit notification templates use the following request:

PUT /messaging/notification_templates/:id.xml
PUT /messaging/notification_templates/:id.json

XML Request Example

```bash
curl -i -X PUT -u user:userpass --url http://onapp.test/messaging/notification_templates/8.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<messaging_notification_template><name>ewpaczin</name><template>HI %{name}! I'm working %{message} </template></messaging_notification_template>
```

JSON Request Example

```bash
```

Where:

- `name` - the name of the template
- `template` - the text of the template

58.4.4 Restore Notification Template to Default
To restore notification template to default, use the following request:

GET /messaging/notification_templates/:id/default.xml
GET /messaging/notification_templates/:id/default.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

58.4.5 Delete Notification Template
To delete notification template, use the following request:

DELETE /messaging/notification_templates/:id.xml
DELETE /messaging/notification_templates/:id.json

XML Request Example

curl -i -X DELETE -u user:userpass --url
    http://onapp.test/messaging/notification_templates/1.xml
    -H 'Accept: application/xml'
    -H 'Content-type: application/xml'

JSON Request Example

curl -i -X DELETE -u user:userpass --url
    http://onapp.test/messaging/notification_templates/1.json
    -H 'Accept: application/json'
    -H 'Content-type: application/json'

58.5 Subscriptions

Subscriptions allow you to determine who of your users are notified about which events. A subscription is the final step of the notifications configuration process which ties together a recipients list, a gateway and a notification template.

58.5.1 Get List of Subscriptions

To get the list of subscriptions, use the following request:

GET /messaging/subscriptions.xml
GET /messaging/subscriptions.json

XML Request Example

curl -i -X GET -H 'Accept: application/xml'
    -H 'Content-type: application/xml'
    -u user:userpass --url
    http://onapp.test/messaging/subscriptions.xml

JSON Request Example

curl -i -X GET -H 'Accept: application/json'
    -H 'Content-type: application/json'
    -u user:userpass --url
    http://onapp.test/messaging/subscriptions.json

XML Output Example
<messaging_subscriptions type="array">
  <messaging_subscription>
    <id type="integer">1</id>
    <name>qaav</name>
    <created_at type="dateTime">2017-01-11T14:39:12+00:00</created_at>
    <updated_at type="dateTime">2017-01-11T14:39:12+00:00</updated_at>
  </messaging_subscription>
  ...
</messaging_subscriptions>

Where:

- **id** - the ID of the subscription
- **name** - the name of the subscription
- **created_at** - the date when the subscription was created in the [YYYY][MM][DD][hh][mm][ss] format
- **updated_at** - the date when the subscription was updated in the [YYYY][MM][DD][hh][mm][ss] format

### 58.5.2 Add Subscription

To add new subscriptions, use the following request:

**POST** /messaging/subscriptions.xml

**POST** /messaging/subscriptions.json

**XML Request Example**

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/messaging/subscriptions.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<messaging_subscription><name>ttwmqtxh</name><recipients_list_ids type="array"><recipients_list_id>21</recipients_list_id></recipients_list_ids><event_type_notification_template_attributes type="array"><event_type_id>1</event_type_id><notification_template_id>9</notification_template_id><event_type_id>2</event_type_id><notification_template_id>10</notification_template_id></event_type_notification_template_attributes><gateway_ids type="array"><gateway_id>17</gateway_id></gateway_ids></messaging_subscription>'
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/messaging/subscriptions.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"messaging_subscription": {"name": "ttwmqtxh", "recipients_list_ids": [21], "event_type_notification_template_attributes": [{"event_type_id": 1, "notification_template_id": 9}, {"event_type_id": 2, "notification_template_id": 10}], "gateway_ids": [17]}}'
```

Where:

- **name** - the name of the subscription
- **recipients_list_ids** - the IDs of the recipient lists you want to add
**event_type_id** - the ID of the event

**notification_template_id** - the ID of the notification template

**gateway_ids** - the IDs of the gateways

### 58.5.3 Edit Subscription

To edit subscriptions, use the following request:

PUT /messaging/subscriptions/:id.xml

PUT /messaging/subscriptions/:id.json

**XML Request Example**

```
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass --url http://onapp.test/messaging/subscriptions/2.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '##json##'
```

**Where:**

**name** - the name of the subscription

**recipients_list_ids** - the IDs of the recipient lists you want to edit

**event_type_id** - the ID of the event

**notification_template_id** - the ID of the notification template

**gateway_ids** - the IDs of the gateways

### 58.5.4 Delete Subscription

To delete subscriptions, use the following request:

DELETE /messaging/subscriptions/:id.xml

DELETE /messaging/subscriptions/:id.json

**XML Request Example**
58.6 Gateways

Gateways are used when setting up a subscription and determine in what way users will be contacted: via email or internal notifications in CP. You can create multiple gateways to verify without any limitation.

58.6.1 Get List of Gateways

To get the list of gateways, use the following request:

GET /messaging/gateways.xml
GET /messaging/gateways.json

XML Request Example

```bash
```

JSON Request Example

```bash
```
<messaging_gateways type="array">
  <messaging_gateway>
    <id type="integer">1</id>
    <name>System SENDMAIL Gateway</name>
    <delivery_method>SENDMAIL</delivery_method>
    <options>
      <from>app@onapp.com</from>
      <host>onapp.com</host>
    </options>
    <created_at type="dateTime">2016-10-28T15:27:16+00:00</created_at>
    <updated_at type="dateTime">2018-01-09T02:53:43+00:00</updated_at>
    <primary type="boolean">true</primary>
  </messaging_gateway>
</messaging_gateways>

Where:
id - the ID of the gateway
name - the name of the gateway
delivery_method - gateway’s delivery method. Can be sendmail, internal or SMTP
from - the email address from which emails will be sent
host - the server IP or URL
created_at - the date in the [YYYY][MM][DD][hh][mm][ss] format
updated_at - the date when the template was updated in the [YYYY][MM][DD][hh][mm][ss] format
primary - the default gateway to send system notifications

58.6.2 Add Internal Gateway

To add an internal gateway, use the following request:

POST /messaging/gateways.xml

POST /messaging/gateways.json

XML Request Example

curl -i -X POST -u user:userpass --url
'Content-type: application/xml' -d
'<messaging_gateway><name>yxkefbmd</name><delivery_method>INTERNAL</delivery_method><options/></options></messaging_gateway>'

JSON Request Example

curl -i -X POST -u user:userpass --url
'Content-type: application/json' -d '{"messaging_gateway": {'"name":
"yxkefbmd", "delivery_method": "INTERNAL", "options": {}}}'

Where:
name - the name of the gateway
**delivery_method** - gateway's delivery method. Can be sendmail, internal or SMTP

### 58.6.3 Add Sendmail Gateway

To add sendmail gateways, use the following request:

**POST /messaging/gateways.xml**

**POST /messaging/gateways.json**

**XML Request example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/messaging/gateways.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<messaging_gateway><name>osipauaw</name><delivery_method>SENDMAIL</delivery_method><primary>true</primary><options><from>test@onapp.com</from><host>localhost.bak</host></messaging_gateway>'
```

**JSON Request example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/messaging/gateways.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"messaging_gateway": {"name": "osipauaw", "delivery_method": "SENDMAIL", "primary": true, "options": {"from": "test@onapp.com", "host": "localhost.bak"}}}'
```

**Where:**

- **name** - the name of the gateway
- **delivery_method** - gateway's delivery method. Can be sendmail, internal or SMTP
- **primary** - the default gateway to send system notifications
- **from** - the email address from which emails will be sent
- **host** - the server IP or URL

### 58.6.4 Add SMTP Gateway

To add an SMTP gateway, use the following request:

**POST /messaging/gateways.xml**

**POST /messaging/gateways.json**

**XML Request Example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/messaging/gateways.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<messaging_gateway><name>uzgfkkbr</name><delivery_method>SMTP</delivery_method><primary>true</primary><options><from>test@onapp.com</from><host=localhost.bak</host><smtp_address>smtp.host.com</smtp_address><smtp_port>22</smtp_port><smtp_domain=localhost</smtp_domain><smtp_user_name>smtp</smtp_user_name><smtp_password>testdsffff</smtp_password><smtp_authentication>plain</smtp_authentication><smtp_enable_starttls_auto>true</smtp_enable_starttls_auto></options></messaging_gateway>'
```

**Where:**

- **name** - the name of the gateway
- **delivery_method** - gateway's delivery method. Can be sendmail, internal or SMTP
- **primary** - the default gateway to send system notifications
JSON Request Example

```bash
curl -i -X POST -u user:userpass --url
  http://onapp.test/messaging/gateways.json -H 'Accept: application/json'
  -H 'Content-type: application/json'
  -d '{"messaging_gateway": {"name": "uzgfkbr", "delivery_method": "SMTP", "primary":true, "options": {"from": "test@onapp.com", "host": "localhost.bak", "smtp_address": "smtp.host.com", "smtp_port": "22", "smtp_domain": "localhost", "smtp_user_name": "smtp", "smtp_password": "testsfff", "smtp_authentication": "plain", "smtp_enable_starttls_auto": "1"}}}'
```

Where:

- **name** - the name of the gateway
- **delivery_method** - gateway’s delivery method. Can be sendmail, internal or SMTP
- **primary** - the default gateway to send system notifications
- **options** - the array of
  - **from** - the email address from which emails will be sent
  - **host** - the server IP or URL
  - **port** - the port of the SMTP server
  - **smtp_domain** - the associated domain
  - **smtp_user_name** - the username to login into SMTP server
  - **smtp_password** - the password to login into SMTP server
  - **smtp_authentication** - select an authentication mechanism: plain, login or cram_md5
  - **smtp_enable_starttls_auto** - enable the StartTLS extension

58.6.5 Delete Gateway

To delete a gateway, use the following request:

DELETE /messaging/gateways/:id.xml
DELETE /messaging/gateways/:id.json

XML Request Example

```bash
curl -i -X DELETE -u user:userpass --url
  -H 'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X DELETE -u user:userpass --url
  -H 'Content-type: application/json'
```
58.7 Get List of Deliveries

To get the list of deliveries, use the following request:

GET /messaging/deliveries.xml
GET /messaging/deliveries.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<messaging_deliveries type="array">
  <messaging_delivery>
    <id type="integer">24095</id>
    <recipient_id type="integer">1</recipient_id>
    <message_id type="integer">65057</message_id>
    <destination>INTERNAL</destination>
    <subscriber_name>John Smith</subscriber_name>
    <created_at type="dateTime">2018-03-16T12:24:29+00:00</created_at>
    <updated_at type="dateTime">2018-03-16T12:24:32+00:00</updated_at>
    <status>complete</status>
    <output>SUCCESS</output>
    <subscription_topic_notification_template_id type="integer">2</subscription_topic_notification_template_id>
    <gateway_id type="integer">2</gateway_id>
  </messaging_delivery>
</messaging_deliveries>
```

**Where:**

- id - the ID of the delivery
- recipient_id - the ID of the recipient
- message_id - the ID of the message
- destination - the destination to which the notification was delivered: SMTP or SENDMAIL for email notifications and INTERNAL for notifications in CP
- subscriber_name - the user to whom the notification is to be delivered
- created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- updated_at - the date when the template was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- status - whether the notification was delivered. This status can indicate that the delivery is Complete, Running or Failed.
output - message about the delivery status
subscription_name - the name of the subscription related to the event
subscription_topic_notification_template_id - the ID of the notification template related to the subscription
gateway_id - the ID of the gateway

58.8 Configuration

Via Configuration section, you can enable notifications for your cloud. All notification sections will be available in the Control Panel if notifications are disabled for the cloud, but no notifications will be sent. Also, you can set the number of notifications to show here.

58.8.1 Enable/Disable Notifications

To enable or disable notifications, use the following request:
PUT /settings.xml?restart=1
PUT /settings.json?restart=1

XML Request Example

curl -i -X PUT -u user:userpass --url
'<<configuration><enable_notifications>true</enable_notifications></configuration>''

JSON Request Example

curl -i -X PUT -u user:userpass --url

Where:

enable_notifications - true if notifications are enabled, otherwise, false

58.8.2 Set Number of Notifications to Show

To set the number of notifications to show, use the following request:
PUT /settings.xml?restart=1
PUT /settings.xml?restart=1

XML Request Example

curl -i -X PUT -u user:userpass --url
'<<configuration><number_of_notifications_to_show>87</number_of_notifications_to_show></configuration>''
JSON Request Example

```
```

Where:
- `configuration` - the array of parameters for notifications
- `number_of_notifications_to_show` - the number of notifications to show

### 58.8.3 Set Notification Prefix

To set notification prefix, use the following request:

PUT /settings.xml?restart=1
PUT /settings.xml?restart=1

**XML Request Example**

```
curl -i -X PUT -u user:userpass --url http://onapp.test/settings.xml?restart=1 -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<configuration><notification_subject_prefix>f19bef8ae95148378eae9f122a00782b</notification_subject_prefix></configuration>'
```

### JSON Request Example

```
curl -i -X PUT -u user:userpass --url http://onapp.test/settings.json?restart=1 -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"configuration": { "notification_subject_prefix": "f19bef8ae95148378eae9f122a00782b"}}'}
```

Where:
- `configuration` - the array of parameters for notifications
- `notification_subject_prefix` - the notification subject prefix
59 OnApp Engine

OnApp engine starts/stops/reloads/checks status of all background tasks that run in the system.

59.1 Get OnApp Engine Status

To get OnApp Engine status, use the following request:

GET /sysadmin_tools/daemon/status.xml
GET /sysadmin_tools/daemon/status.json

XML Request Example

```
```

JSON Request Example

```
curl -i -X GET -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json'
http://onapp.test/sysadmin_tools/daemon/status.json
```

To get the status of the OnApp Engine on a particular node, add the node’s IP as a parameter. This request applies to clouds with High Availability enabled.

XML Request Example

```
curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml'
http://onapp.test/sysadmin_tools/daemon/status.xml -d '<node>199.169.1.203</node>'
```

JSON Request Example

```
curl -i -X GET -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json'
http://onapp.test/sysadmin_tools/daemon/status.json -d '{"node": "199.169.1.203"}'
```

XML Response Example

```
<daemon>
<status>Online</status>
<ip>138.0.0.2</ip>
</daemon>
```

Where:
**status** - OnApp Engine status: Online or Offline

**ip** - IP address of the node on which the OnApp engine is running

### 59.2 Start OnApp Engine

To start OnApp engine, use the following request:

```
POST /sysadmin_tools/daemon/start.xml
POST /sysadmin_tools/daemon/start.json
```

**XML Request Example**

```
```

**JSON Request Example**

```
```

To start the OnApp Engine on a particular node add the node's IP as a parameter. This request applies to clouds with High Availability enabled.

**XML Request Example**

```
```

**JSON Request Example**

```
```

### 59.3 Stop OnApp Engine

To stop OnApp engine, use the following request:

```
POST /sysadmin_tools/daemon/stop.xml
POST /sysadmin_tools/daemon/stop.json
```

**XML Request Example**
To stop the OnApp Engine on a particular node add the node’s IP as a parameter. This request applies to clouds with High Availability enabled.

**XML Request Example**


**JSON Request Example**


### 59.4 Reload OnApp Engine

To reload a background task daemon, use the following request:

POST /sysadmin_tools/daemon/reload.xml
POST /sysadmin_tools/daemon/reload.json

**XML Request Example**


**JSON Request Example**


To reload the OnApp Engine on a particular node add the node’s IP as a parameter. This request applies to clouds with High Availability enabled.
XML Request Example

```bash
```

JSON Request Example

```bash
```
60 OVAs

The OVA import functionality allows you to deploy to OnApp virtual servers created at other virtualization platforms. OVA is a file with the Open Virtualization Format (OVF) package contents all zipped into a single archive. OVF is an open-source standard for packaging and distributing software applications for virtual servers.

60.1 Get List of OVAs

To view the list of OVAs, use the following request:

GET /template_ovas.xml
GET /template_ovas.json

XML Request Example

```bash
curl -i -X GET -u user:userpass --url http://onapp.test/template_ovas.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass --url http://onapp.test/template_ovas.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

XML Output Example
<image_template_ova>
  <id type="integer">103</id>
  <label>centos6.7</label>
  <created_at type="dateTime">2017-04-05T14:38:41+03:00</created_at>
  <updated_at type="dateTime">2017-05-29T13:59:47+03:00</updated_at>
  <version>1.0</version>
  <file_name>ufwfmvuhloyjsv.tar.gz</file_name>
  <operating_system>other</operating_system>
  <operating_system_distro>other</operating_system_distro>
  <allowed_swap type="boolean">false</allowed_swap>
  <state>active</state>
  <checksum nil="true"/>
  <allow_resize_without_reboot nil="true"/>
  <user_id nil="true"/>
  <template_size type="integer">228344</template_size>
  <operating_system_arch>x64</operating_system_arch>
  <operating_system_edition nil="true"/>
  <operating_system_tail nil="true"/>
  <parent_template_id nil="true"/>
  <virtualization type="array">
    <virtualization>kvm</virtualization>
  </virtualization>
  <min_memory_size type="integer">1024</min_memory_size>
  <disk_target_device nil="true"/>
  <cdn type="boolean">false</cdn>
  <backup_server_id type="integer">1</backup_server_id>
  <ext4 type="boolean">false</ext4>
  <smart_server type="boolean">false</smart_server>
  <baremetal_server type="boolean">false</baremetal_server>
  <initial_password>Password1</initial_password>
  <initial_username>root</initial_username>
  <remote_id nil="true"/>
  <manager_id>ufwfmvuhloyjsv</manager_id>
  <resize_without_reboot_policy nil="true"/>
  <application_server type="boolean">false</application_server>
  <draas type="boolean">false</draas>
  <properties>
    <cpus type="integer">1</cpus>
    <vmdks type="array">
      <vmdk>centos6.7-disk1.vmdk</vmdk>
    </vmdks>
    <uuid>96b1b976-1e74-492f-921e-27a0e4a1eff6</uuid>
  </properties>
</image_template_ova>

Where:

- **id** - the ID of the OVA file
- **label** - the title of the OVA file
- **created_at** - the time when the OVA template was created, in [YYYY][MM][DD][hh][mm][ss]Z
- **updated_at** - the time when the OVA template was updated, in [YYYY][MM][DD][hh][mm][ss]Z
- **version** - the version of the OVA file
- **file_name** - the name of the OVA tar.gz file
operating_system - the operating system of OVA
operating_system_distro - the operating system distribution of OVA
state - the status of the OVA template
template_size - the size of the OVA template
allowed_hot_migrate - true if hot migration for VS created from this OVA is enabled, otherwise, false
operating_system_arch - the architecture of the OVA file (x86 or x64)

virtualization - the virtualization type of the converted OVA template (kvm or vcenter)
min_memory_size - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.
backup_server_id - the ID of the backup server where OVA is stored

initial_password - the initial password for vCenter-based OVA templates
initial_username - the initial username for vCenter-based OVA templates
properties - the attributes of OVA template:
- cpus - the amount of OVA CPUs
- vmdks - the label of a vmdk disk
- grub - the GRUB version
- uuid - the ID of the OVA template in the database
type - the type of the image template

The following output parameters do not currently apply to OVAs:
allowed_swap
checksum
allow_resize_without_reboot
user_id
operating_systemEdition
operating_systemTail
parent_template_id
disk_target_device
cdn
ext4
smart_server
baremetal_server
remote_id
manager_id
resize_without_reboot_policy
application_server
draas
locked
Page History
v.6.3 Edge 1
- added grub parameter

v.5.7
- moved the following parameters to those that can be applied to OVAs:
  - operating_system_arch
  - initial_password
  - initial_username

v.5.5
- added the allowed_hot_migrate parameter
- removed the min_disk_size parameter

60.2 Get List of System OVAs
To view the list of system OVAs, use the following request:
GET /template_ovas/system.xml
GET /template_ovas/system.json

XML Request Example

curl -i -X GET -u user:userpass --url
'Content-type: application/xml'

JSON Request Example

curl -i -X GET -u user:userpass --url
-H 'Content-type: application/json'

XML Output Example
<image_template_ovas type="array">
  <image_template_ova>
    <id type="integer">103</id>
    <label>centos6.7</label>
    <created_at type="dateTime">2017-04-05T14:38:41+03:00</created_at>
    <updated_at type="dateTime">2017-05-29T13:59:47+03:00</updated_at>
    <version>1.0</version>
    <file_name>ufwfmvuhloyjsv.tar.gz</file_name>
    <operating_system>other</operating_system>
    <operating_system_distro>other</operating_system_distro>
    <allowed_swap type="boolean">false</allowed_swap>
    <state>active</state>
    <checksum nil="true"/>
    <allow_resize_without_reboot nil="true"/>
    <user_id nil="true"/>
    <template_size type="integer">228344</template_size>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <virtualization type="array">
      <virtualization>kvm</virtualization>
    </virtualization>
    <min_memory_size type="integer">1024</min_memory_size>
    <disk_target_device nil="true"/>
    <cdn type="boolean">false</cdn>
    <backup_server_id type="integer">1</backup_server_id>
    <ext4 type="boolean">false</ext4>
    <baremetal_server type="boolean">false</baremetal_server>
    <initial_password>Password1</initial_password>
    <initial_username>root</initial_username>
    <remote_id nil="true"/>
    <manager_id>ufwfmvuhloyjsv</manager_id>
    <resize_without_reboot_policy nil="true"/>
    <application_server type="boolean">false</application_server>
    <draas type="boolean">false</draas>
    <properties>
      <cpus type="integer">1</cpus>
      <vmdks type="array">
        <vmdk>centos6.7-disk1.vmdk</vmdk>
      </vmdks>
      <uuid>96b1b976-1e74-492f-921e-7a0e4a1eff6</uuid>
    </properties>
  </image_template_ova>
</image_template_ova>
</image_template_ovas>

Where:

- **id** - the ID of the OVA file
- **label** - the title of the OVA file
- **created_at** - the time when the OVA template was created, in [YYYY][MM][DD][hh][mm][ss]Z
- **updated_at** - the time when the OVA template was updated, in [YYYY][MM][DD][hh][mm][ss]Z
- **version** - the version of the OVA file
- **file_name** - the name of the OVA tar.gz file
**operating_system** - the operating system of OVA
**operating_system_distro** - the operating system distribution of OVA
**state** - the status of the OVA template
**template_size** - the size of the OVA template
**allowed_hot_migrate** - true if hot migration for VS created from this OVA is enabled, otherwise, false
**operating_system_arch** - the architecture of the OVA file (x86 or x64)
**virtualization** - the virtualization type of the converted OVA template (kvm or vcenter)
**min_memory_size** - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.
**backup_server_id** - the ID of the backup server where OVA is stored

**initial_password** - the initial password for vCenter-based OVA templates
**initial_username** - the initial username for vCenter-based OVA templates
**properties** - the attributes of OVA template:
- **cpus** - the amount of OVA CPUs
- **vmdks** - the label of a vmdk disk
- **grub** - the GRUB version
- **uuid** - the ID of the OVA template in the database
**type** - the type of the image template

The following output parameters do not currently apply to OVAs:
**allowed_swap**
**checksum**
**allow_resize_without_reboot**
**user_id**
**operating_system_edition**
**operating_system_tail**
**parent_template_id**
**disk_target_device**
**cdn**
**ext4**
**smart_server**
**baremetal_server**
**remote_id**
**manager_id**
**resize_without_reboot_policy**
**application_server**
**draas**
**locked**
Page History
v.6.3 Edge 1
• added grub parameter

v.5.7
• moved the following parameters to those that can be applied to OVAs:
  o operating_system_arch
  o initial_password
  o initial_username

v.5.5
• added the allowed_hot_migrate parameter
• removed the min_disk_size parameter

60.3 Get List of Own OVAs
To view the list of own OVAs, use the following request:
GET /template_ovas/own.xml
GET /template_ovas/own.json

XML Request Example
```
```

JSON Request Example
```
```

XML Output Example
Where:

- **id** - the ID of the OVA file
- **label** - the title of the OVA file
- **created_at** - the time when the OVA template was created, in [YYYY][MM][DD]T[hh][mm][ss]Z
- **updated_at** - the time when the OVA template was updated, in [YYYY][MM][DD]T[hh][mm][ss]Z
- **version** - the version of the OVA file
- **file_name** - the name of the OVA tar.gz file
- **operating_system** - the operating system of OVA
operating_system_distro - the operating system distribution of OVA
state - the status of the OVA template
template_size - the size of the OVA template
allowed_hot_migrate - true if hot migration for VS created from this OVA is enabled, otherwise, false
operating_system_arch - the architecture of the OVA file (x86 or x64)

virtualization - the virtualization type of the converted OVA template (kvm or vcenter)
min_memory_size - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.
backup_server_id - the ID of the backup server where OVA is stored

initial_password - the initial password for vCenter-based OVA templates
initial_username - the initial username for vCenter-based OVA templates

properties - the attributes of OVA template:
  • cpus - the amount of OVA CPUs
  • vmdks - the label of a vmdk disk
  • grub - the GRUB version
  • uuid - the ID of the OVA template in the database

The following output parameters do not currently apply to OVAs:
allowed_swap
checksum
allow_resize_without_reboot
user_id
operating_system_edition
operating_system_tail
parent_template_id
disk_target_device
cdn
ext4
smart_server
baremetal_server
remote_id
manager_id
resize_without_reboot_policy
application_server
draas
locked
Page History
v.6.3 Edge 1
- added grub parameter

v.5.7
- moved the following parameters to those that can be applied to OVAs:
  - operating_system_arch
  - initial_password
  - initial_username

v.5.5
- added the allowed_hot_migrate parameter
- removed the min_disk_size parameter

60.4 Get List of User OVAs

To view the list of user OVAs, use the following request:
GET /template_ovas/user.xml
GET /template_ovas/user.json

XML Request Example
```
```

JSON Request Example
```
```

XML Output Example
<image_template_ovas type="array">
  <image_template_ova>
    <id type="integer">103</id>
    <label>centos6.7</label>
    <created_at type="dateTime">2017-04-05T14:38:41+03:00</created_at>
    <updated_at type="dateTime">2017-05-29T13:59:47+03:00</updated_at>
    <version>1.0</version>
    <file_name>ufwfvmvuhlojsv.tar.gz</file_name>
    <allow_swap type="boolean">false</allow_swap>
    <state>active</state>
    <checksum nil="true"/>
    <allow_resize_without_reboot nil="true"/>
    <template_size type="integer">228344</template_size>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <operating_system_arch>x64</operating_system_arch>
    <min_memory_size type="integer">1024</min_memory_size>
    <disk_target_device nil="true"/>
    <smart_server type="boolean">false</smart_server>
    <baremetal_server type="boolean">false</baremetal_server>
    <initial_password>Password1</initial_password>
    <initial_username>root</initial_username>
    <manager_id>ufwfvmvuhlojsv</manager_id>
    <allow_resize_without_reboot_policy nil="true"/>
    <application_server type="boolean">false</application_server>
    <draas type="boolean">false</draas>
    <properties>
      <cpus type="integer">1</cpus>
      <vmdks type="array">
        <vmdk>centos6.7-disk1.vmdk</vmdk>
      </vmdks>
      <uuid>96b1b976-1e74-492f-921e-27a0e4a1eef6</uuid>
    </properties>
    <locked type="boolean">false</locked>
  </image_template_ova>
  <image_template_ova>... </image_template_ova>
</image_template_ovas>

Where:

- **id** - the ID of the OVA file
- **label** - the title of the OVA file
- **created_at** - the time when the OVA template was created, in [YYYY][MM][DD][T][hh][mm][ss]Z
- **updated_at** - the time when the OVA template was updated, in [YYYY][MM][DD][T][hh][mm][ss]Z
- **version** - the version of the OVA file
- **file_name** - the name of the OVA tar.gz file
- **operating_system** - the operating system of OVA
operating_system_distro - the operating system distribution of OVA
state - the status of the OVA template
template_size - the size of the OVA template
allowed_hot_migrate - true if hot migration for VS created from this OVA is enabled, otherwise, false
operating_system_arch - the architecture of the OVA file (x86 or x64)
virtualization - the virtualization type of the converted OVA template (kvm or vcenter)
min_memory_size - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.
backup_server_id - the ID of the backup server where OVA is stored
initial_password - the initial password for vCenter-based OVA templates
initial_username - the initial username for vCenter-based OVA templates
properties - the attributes of OVA template:
  • cpus - the amount of OVA CPUs
  • vmdks - the label of a vmdk disk
  • grub - the GRUB version
  • uuid - the ID of the OVA template in the database
type - the type of the image template

The following output parameters do not currently apply to OVAs:
allowed_swap
checksum
allow_resize_without_reboot
user_id
operating_system_edition
operating_system_tail
parent_template_id
disk_target_device
cdn
ext4
smart_server
baremetal_server
remote_id
manager_id
resize_without_reboot_policy
application_server
draas
locked
Page History
v.6.3 Edge 1
- added grub parameter

v.5.7
- moved the following parameters to those that can be applied to OVAs:
  - operating_system_arch
  - initial_password
  - initial_username

v.5.5
- added the allowed_hot_migrate parameter
- removed the min_disk_size parameter

60.5 Get OVA Details
To view the details of the particular OVA, use the following request:
GET /template_ovas/:id.xml
GET /template_ovas/:id.json

XML Request Example
```
curl -i -X GET -u user:userpass --url
'Content-type: application/xml'
```

JSON Request Example
```
curl -i -X GET -u user:userpass --url
'Content-type: application/json'
```

XML Output Example
Where:

- **id** - the ID of the OVA file
- **label** - the title of the OVA file
- **created_at** - the time when the OVA template was created, in `YYYY-MM-DDThh:mm:ssZ`
- **updated_at** - the time when the OVA template was updated, in `YYYY-MM-DDThh:mm:ssZ`
- **version** - the version of the OVA file
- **file_name** - the name of the OVA tar.gz file
- **operating_system** - the operating system of OVA
- **operating_system_distro** - the operating system distribution of OVA
state - the status of the OVA template

template_size - the size of the OVA template

allowed_hot_migrate - true if hot migration for VS created from this OVA is enabled, otherwise, false

operating_system_arch - the architecture of the OVA file (x86 or x64)

virtualization - the virtualization type of the converted OVA template (kvm or vcenter)

min_memory_size - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.

backup_server_id - the ID of the backup server where OVA is stored

initial_password - the initial password for vCenter-based OVA templates

initial_username - the initial username for vCenter-based OVA templates

properties - the attributes of OVA template:

- cpus - the amount of OVA CPUs
- vmdks - the label of a vmdk disk
- grub - the GRUB version
- uuid - the ID of the OVA template in the database

- type - the type of the image template

The following output parameters do not currently apply to OVAs:

allowed_swap
checksum
allow_resize_without_reboot
user_id
operating_system_edition
operating_system_tail
parent_template_id
disk_target_device
cdn
ext4
smart_server
baremetal_server
remote_id
manager_id
resize_without_reboot_policy
application_server
draas
locked
Page History
v.6.3 Edge 1
- added grub parameter
v.5.7
- moved the following parameters to those that can be applied to OVAs:
  o operating_system_arch
  o initial_password
  o initial_username
v.5.5
- added the allowed_hot_migrate parameter
- removed the min_disk_size parameter

60.6 Get OVA Disks

To view the details of a particular OVA disk, use the following request:

GET /template_ovas/:id/disks.xml
GET /template_ovas/:id/disks.json

XML Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/template_ovas/2/disks.xml
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/template_ovas/2/disks.json
-H 'Accept: application/json'
-H 'Content-type: application/json'
```

XML Output Example

```
<image_template_disks type="array">
  <image_template_disk>
    <id type="integer">2</id>
    <identifier>identifier1</identifier>
    <template_id type="integer">2</template_id>
    <disk_size type="integer">10</disk_size>
    <disk_type type="integer">0</disk_type>
    <mount_point nil="true"/>
    <label>L3</label>
    <file_system>ext3</file_system>
    <mounted type="boolean">false</mounted>
  </image_template_disk>
  ...
</image_template_disks>
```
Where:

- `image_template_disks` - the array of OVA disks
- `image_template_disk` - the array of the disk parameters
- `id` - the ID of the disk
- `identifier` - the identifier of the disk
- `template_id` - the ID of the OVA template
- `disk_size` - the size of the disk in GB
- `mount_point` - the physical location in the partition used as a root filesystem
- `label` - the label of the disk
- `file_system` - the file system of the disk. For Linux-based virtual servers, the `ext3`, `ext4`, and `xfs` filesystems are available. For Windows-based virtual servers, only `ntfs` is available.
- `mounted` - true if the disk inside OS is mounted automatically, otherwise, `false`
- `created_at` - the date when the disk was created in the `[YYYY][MM][DD][T][hh][mm][ss]Z` format
- `updated_at` - the date when the disk was updated in the `[YYYY][MM][DD][T][hh][mm][ss]Z` format

### 60.7 Upload OVA

To add a new OVA, use the following request:

```bash
POST /template_ova.xml
POST /template_ova.json
```

**XML Request Example**

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/template_ova.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'&lt;image_template_ova&gt;&lt;label&gt;MyOVA&lt;/label&gt;&lt;version&gt;7.5&lt;/version&gt;&lt;min_memory_size&gt;512&lt;/min_memory_size&gt;&lt;backup_server_id&gt;2&lt;/backup_server_id&gt;&lt;file_url&gt;https://sourceforge.net/projects/virtualappliances/files/Linux/CentOS/CentOS-6.4-i386-minimal.ova&lt;/file_url&gt;&lt;/image_template_ova&gt;
'
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass --url
'Content-type: application/json' -d '{"image_template_ova": {"label": "MyOVA","version": "7.5","min_memory_size": "512","backup_server_id": "2","file_url": "https://sourceforge.net/projects/virtualappliances/files/Linux/CentOS/CentOS-6.4-i386-minimal.ova"}}'
```

**Where:**

- `label*` - the title of the OVA file
version* - the version of the OVA file

min_memory_size* - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.

backup_server_id* - the ID of the backup server where OVA is stored

declaration file_url* - the URL from which the OVA file can be uploaded

Page History

v.5.7

- removed the following parameters:
  - operating_system
  - operating_system_distro
  - make_public
  - allowed_hot_migrate
  - virtualization

v.5.5

- added the allowed_hot_migrate parameter
- removed the min_disk_size parameter

60.8 Convert OVA

To convert the OVA template into a virtualization format, use the following request:

POST /template_ovas/:id/converting.xml

POST /template_ovas/:id/converting.json

XML Request Example

curl -i -X POST -u user:userpass --url http://onapp.test/template_ovas/12/converting.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<image_template_ova><label>MyOVA</label><virtualization>vcenter</virtualization><operating_system>linux</operating_system><operating_system_distro>ubuntu</operating_system_distro><initial_username>username</initial_username><initial_password>admin</initial_password><allowed_hot_migrate>true</allowed_hot_migrate><make_public>true</make_public><operating_system_arch>x64</operating_system_arch></image_template_ova>'

JSON Request Example

curl -i -X POST -u user:userpass --url http://onapp.test/template_ovas/12/converting.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"image_template_ova":{"label": "MyOVA", "virtualization": "kvm", "operating_system": "linux", "operating_system_distro": "ubuntu", "allowed_hot_migrate": "true", "make_public": "true", "operating_system_arch": "x64"}}'

Where:
**label** - the title of the OVA file

**virtualization** - the virtualization type of the converted OVA template (kvm, vcenter or kvm_virtio)

**operating_system** - the operating system of OVA

**operating_system_distro** - the operating system distribution of OVA

**initial_password** - the password for the guest operating system for vCenter-based OVA templates

**initial_username** - the username for the guest operating system for vCenter-based OVA templates

**allowed_hot_migrate** - true if hot migration for VS created from this OVA is enabled, otherwise, false

**make_public** - true for the OVA file to be available to all users, otherwise, false

**operating_system_arch** - the architecture of the OVA file (x86 or x64)

**operating_system_edition** - the edition of the OVA (required for Windows-based OVAs only)

**operating_system_tail** - the updated release of Windows OS distribution. If updated, then the parameter is R2, otherwise, empty (for Windows-based OVAs only).

### 60.9 Unlock OVA

The OVA file is locked for the time period while it is being converted. You can unlock OVA to make the following actions instantly available: make public, edit OVA, delete OVA or delete OVA files.

To unlock the OVA file, use the following request:

POST /template_ova/:id/unlock.xml

POST /template_ova/:id/unlock.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where:

**id** - the identification number of the OVA file that you want to unlock

If the OVA file is unlocked successfully, the HTTP 201 response is returned.
60.10 Edit OVA

To update an OVA file, use the following request:

PUT /template_ovas/:id.xml
PUT /template_ovas/:id.json

XML Request Example

```bash
```

JSON Request Example

```bash
curl -i -X PUT -u user:userpass --url http://onapp.test/template_ovas/9.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"image_template_ova": {"label": "MyOVA","version": "8.5","min_memory_size": "512","allowed_hot_migrate": "true"}}'
```

Where:

- **label** - the title of the OVA file
- **version** - the version of the OVA file
- **min_memory_size** - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.
- **allowed_hot_migrate** - set true if hot migration for VS created from this OVA is enabled, otherwise, false
- **initial_password** - the initial password for vCenter-based OVA templates
- **initial_username** - the initial username for vCenter-based OVA templates

Page History

v.5.7

- removed the following parameters:
  - operating_system
  - operating_system_distro
  - virtualization
  - file_url
  - make_public

- added the following parameters:
  - initial_password
60.11 Make OVA Public

To make the OVA file public, use the following request:

POST /template_ovas/:id/make_public.xml
POST /template_ovas/:id/make_public.json

XML Request Example

```bash
curl -i -X POST -u user:userpass --url
```

JSON Request Example

```bash
curl -i -X POST -u user:userpass --url
```

Where:

- `id` - the identification number of the OVA file that you want to make public

If the OVA file is queued to be moved to a public list successfully, the HTTP 201 response is returned.

60.12 Search OVA

To apply filtering for OVAs, use the following request:

GET /template_ovas.xml?search_filter[query]=text
GET /template_ovas.json?search_filter[query]=text

XML Request Example

```bash
curl -i -X GET -u user:userpass --url
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass --url
```
<image_template_ova type="array">
  <image_template_ova>
    <id type="integer">103</id>
    <label>centos6.7</label>
    <created_at type="dateTime">2017-04-05T14:38:41+03:00</created_at>
    <updated_at type="dateTime">2017-05-29T13:59:47+03:00</updated_at>
    <version>1.0</version>
    <file_name>ufwfmvuhlojysv.tar.gz</file_name>
    <operating_system>other</operating_system>
    <operating_system_distro>other</operating_system_distro>
    <allowed_swap type="boolean">false</allowed_swap>
    <state>active</state>
    <checksum nil="true"/>
    <allow_resize_without_reboot nil="true"/>
    <user_id nil="true"/>
    <template_size type="integer">228344</template_size>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <virtualization type="array">
      <virtualization>kvm</virtualization>
      <min_memory_size type="integer">1024</min_memory_size>
      <disk_target_device nil="true"/>
      <cdn type="boolean">false</cdn>
      <backup_server_id type="integer">1</backup_server_id>
      <ext4 type="boolean">false</ext4>
      <baremetal_server type="boolean">false</baremetal_server>
      <initial_password>Password1</initial_password>
      <initial_username>root</initial_username>
      <remote_id nil="true"/>
      <manager_id>ufwfmvuhlojysv</manager_id>
      <application_server type="boolean">false</application_server>
      <draas type="boolean">false</draas>
      <properties>
        <cpus type="integer">1</cpus>
        <vmdks type="array">
          <vmdk>centos6.7-disk1.vmdk</vmdk>
        </vmdks>
        <uuid>96b1b976-1e74-492f-921e-27a0e4a1eff6</uuid>
      </properties>
      <locked type="boolean">false</locked>
      <type>ImageTemplateOva</type>
    </virtualization>
  </image_template_ova>
</image_template_ova>

Where:
- id - the ID of the OVA file
- label - the title of the OVA file
- created_at - the time when the OVA template was created, in [YYYY][MM][DD][hh][mm][ss]Z
- updated_at - the time when the OVA template was updated, in [YYYY][MM][DD][hh][mm][ss]Z
- version - the version of the OVA file
- file_name - the name of the OVA tar.gz file
- operating_system - the operating system of OVA
operating_system_distro - the operating system distribution of OVA

state - the status of the OVA template

template_size - the size of the OVA template

allowed_hot_migrate - true if hot migration for VS created from this OVA is enabled, otherwise, false

operating_system_arch - the architecture of the OVA file (x86 or x64)

virtualization - the virtualization type of the converted OVA template (kvm or vcenter)

min_memory_size - the minimum memory size required to build VS from this OVA template (MB). It is not possible to set the memory size less than the RAM set in the OVA file itself.

backup_server_id - the ID of the backup server where OVA is stored

initial_password - the initial password for vCenter-based OVA templates

initial_username - the initial username for vCenter-based OVA templates

properties - the attributes of OVA template:

- cpus - the amount of OVA CPUs
- vmdks - the label of a vmdk disk
- grub - the GRUB version
- uuid - the ID of the OVA template in the database

type - the type of the image template

The following output parameters do not currently apply to OVAs:

allowed_swap
checksum
allow_resize_without_reboot
user_id
operating_system_edition
operating_system_tail
parent_template_id
disk_target_device
cdn
ext4
smart_server
baremetal_server
remote_id
manager_id
resize_without_reboot_policy
application_server
draas
locked
Page History
v.6.3 Edge 1
- added grub parameter

v.5.7
- moved the following parameters to those that can be applied to OVAs:
  - operating_system_arch
  - initial_password
  - initial_username

v.5.5
- added the allowed_hot_migrate parameter
- removed the min_disk_size parameter

60.13 Delete OVA Files
To delete OVA files, use the following request:
POST /template_ovas/:id/delete_files.xml
POST /template_ovas/:id/delete_files.json
XML Request Example
```bash
```

JSON Request Example
```bash
```

Where:
id - the identification number of the OVA which files you want to delete

60.14 Delete OVA
To delete the OVA file from the system, use the following request:
DELETE /template_ovas/:id.xml
DELETE /template_ovas/:id.json
XML Request Example

JSON Request Example


Where:

id - the identification number of the OVA file that you want to delete
61 Pagination

Pagination lets you split large data list into several pages and specify the number of items displayed per page (the example for Users is provided).

To view the specific page, use the following request:

**Example 1**

**XML Request Example**
```
curl -i -X GET -u user:password --url http://onapp.test/users.xml/page/2
```

**JSON Request Example**
```
curl -i -X GET -u user:password --url http://onapp.test/users.json/page/2
```

**Example 2**

**XML Request Example**
```
curl -i -X GET -u user:password --url http://83.170.110.181/users.xml?page=2
```

**JSON Request Example**
```
curl -i -X GET -u user:password --url http://83.170.110.181/users.json?page=2
```

Where you have to specify the URL and the number of page to display.

To specify the number of items displayed per page, use the following request:

**Example 1**

**XML Request Example**
```
curl -i -X GET -u user:password --url http://onapp.test/users.xml/per_page/4
```

**JSON Request Example**
```
curl -i -X GET -u user:password --url http://onapp.test/users.json/per_page/4
```

**Example 2**

**XML Request Example**
Where you have to specify the URL and the number of items to display per page.

To specify the page number and the number of items displayed per page, run one of the following requests:

**Example 1**

**XML Request Example**
```
curl -i -X GET -u user:password --url
http://83.170.110.181/users.xml/page/2/per_page/4
```

**JSON Request Example**
```
curl -i -X GET -u user:password --url
http://83.170.110.181/users.json/page/2/per_page/4
```

**Example 2**

**XML Request Example**
```
curl -i -X GET -u user:password --url
http://83.170.110.181/users.xml/per_page/4/page/2
```

**JSON Request Example**
```
curl -i -X GET -u user:password --url
http://83.170.110.181/users.json/per_page/4/page/2
```

**Example 3**

**XML Request Example**
```
curl -i -X GET -u user:password --url
http://83.170.110.181/users.xml?page=2&per_page=4
```

**JSON Request Example**
Where you'll have to specify the page URL, page number and amount of items displayed per page.

**XML Output Example**

```
curl -i -X GET -u user:password --url
http://83.170.110.181/users.json?page=2&per_page=4
```

```
HTTP/1.1 200 OK
Date: Mon, 25 Jun 2012 09:43:35 GMT
Server: Apache/2.2.3 (CentOS)
X-Powered-By: Phusion Passenger (mod_rails/mod_rack) 3.0.9
X-Total: 37
X-Limit: 2
X-Page: 4
X-UA-Compatible: IE=Edge,chrome=1
ETag: "3eecc8dfb2f935df854612cd39a03f49a"
Cache-Control: must-revalidate, private, max-age=0
X-Request-Id: 7162179190b3f21af71542495d3e72b
X-Runtime: 0.187337
X-Rack-Cache: miss
Set-Cookie: _session_id=4f5fca690dcd4a8ba04af1902305b6; path=/; HttpOnly
Status: 200
Connection: close
Transfer-Encoding: chunked
Content-Type: application/json; charset=utf-8
```

**Where:**

- **X-Total** – the total number of displayed items
- **X-Limit** – the number of items displayed per page
- **X-Page** – the page number
62 Payments

This chapter provides API for payments created in the cloud. It enables you to view the list of payments, create edit or delete payments.

62.1 Get List of All User Payments

To get the list of all user payments, use the following request:
GET /billing/user/payments.xml
GET /billing/user/payments.json

XML Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/billing/user/payments.xml -H 'Accept: application/xml' -
H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X GET -u user:userpass --url
http://onapp.test/billing/user/payments.json -H 'Accept: application/json' -
H 'Content-type: application/json'
```

XML Output Example

```
<payments type="array">
  <payment>
    <id type="integer">1</id>
    <amount type="decimal">2.0</amount>
    <payer_id type="integer">4</payer_id>
    <invoice_number>1</invoice_number>
    <created_at type="datetime">2015-12-02T13:44:24+02:00</created_at>
    <updated_at type="datetime">2015-12-02T13:44:24+02:00</updated_at>
  </payment>
  ...
</payments>
```

Where:

- **id** - payment ID
- **amount** - the amount of money in the currency set in the bucket
- **payer_id** - the ID of the user
- **invoice_number** - the optional number of invoice
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss] format
- **updated_at** - the date in the [YYYY][MM][DD][hh][mm][ss] format

Page History

v. 5.6

- removed the type parameter
62.2 Get List of Payments of Particular User

To get the list of payments of a particular user, use the following request:

GET /billing/user/payments/:id.xml
GET /billing/user/payments/:id.json

XML Request Example

curl -i -X GET -u user:password -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<payer_id>11</payer_id>'

JSON Request Example

curl -i -X GET -u user:password -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"payer_id": "11"}'

XML Output Example

<payments type="array">
  <payment>
    <id type="integer">11</id>
    <amount type="decimal">20.0</amount>
    <payer_id type="integer">11</payer_id>
    <invoice_number>12</invoice_number>
    <created_at type="datetime">2016-02-10T17:42:22+02:00</created_at>
    <updated_at type="datetime">2016-02-10T17:42:22+02:00</updated_at>
  </payment>
  <payment>
    <id type="integer">13</id>
    <amount type="decimal">12.55</amount>
    <payer_id type="integer">11</payer_id>
    <invoice_number>3</invoice_number>
    <created_at type="datetime">2016-02-15T10:43:26+02:00</created_at>
    <updated_at type="datetime">2016-02-15T10:48:13+02:00</updated_at>
  </payment>
</payments>

Where:

id - payment ID

amount - the amount of money in the currency set in the bucket

payer_id - the ID of the user

invoice_number - the optional number of invoice

created_at - the date in the [YYYY][MM][DD][hh][mm][ss] format

updated_at - the date in the [YYYY][MM][DD][hh][mm][ss] format
remove the type parameter

62.3 Create User Payment

To create a payment for a particular user, use the following request:

POST /billing/user/payments.xml
POST /billing/user/payments.json

XML Request Example

curl -i -X POST -u user:userpass --url
'<payment><payer_id>11</payer_id><invoice_number>3</invoice_number><amount>12.55</amount></payment>

JSON Request Example

curl -i -X POST -u user:userpass --url

Where:
payer_id - ID of the user
invoice_number - optional number of the invoice
amount* - amount of the payment (should be higher than zero)

62.4 Edit User Payment

To edit a payment, use the following request:

PUT /billing/user/payments/:id.xml
PUT /billing/user/payments/:id.json

XML Request Example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/payments/5.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'＜payment><payer_id>1</payer_id><invoice_number>1</invoice_number><amount>12.56</amount></payment>

JSON Request Example
curl -i -X PUT -u user:userpass --url

Where:

*id* - the ID of the payment

*payer_id* - ID of the user

*invoice_number* - optional number of the invoice

*amount* - amount of the payment (should be higher than zero)

### 62.5 Delete User Payment

To delete a payment, use the following request:

DELETE /billing/user/payments/:id.xml
DELETE /billing/user/payments/:id.json

**XML Request Example**

curl -i -X DELETE -u user:userpass --url
http://onapp.test/billing/user/payments/5.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request Example**

curl -i -X DELETE -u user:userpass --url
http://onapp.test/billing/user/payments/5.json -H 'Accept: application/json' -H 'Content-type: application/json'
63 Publishing Rules

Publishing rules function as destination NAT, making virtual servers accessible from outside. If the virtual server is running within a customer network, it is necessary to configure to enable Internet access to this virtual server. To publish a VS port, you have to configure a publishing rule for the VS.

63.1 Get List of Publishing Rules

To get the list of publishing rules, use the following request:

GET /virtual_machines/:vm_id/publications.xml
GET /virtual_machines/:vm_id/publications.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<publications type="array">
  <publication>
    <created_at type="datetime">2012-09-14T12:57:49+03:00</created_at>
    <id type="integer">1</id>
    <is_built type="boolean">false</is_built>
    <outside_ip_address_id nil="true"/>
    <port type="integer">80</port>
    <protocol>UDP</protocol>
    <rule_number type="integer">51</rule_number>
    <updated_at type="datetime">2012-09-14T12:57:49+03:00</updated_at>
    <virtual_machine_id type="integer">66</virtual_machine_id>
  </publication>
</publications>
```

Where:

- **publications** - an array of the publishing rules
- **created_at** - the date when the publishing rule was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- **id** - publishing rule ID
- **outside_ip_address_id** - ID of a free IPv4 address
- **port** - the port for which this rule is active
- **protocol** - type of the protocol.
- **rule_number** -
**updated_at** - the date when the publishing rule was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**virtual_machine_id** - ID of the virtual server

### 63.2 Add Publishing Rule

To add a new publishing rule, use the following request:

POST /virtual_machines/:vm_id/publications.xml
POST /virtual_machines/:vm_id/publications.json

**XML Request Example**

```bash
curl -i -X POST -u user:userpass -d 
"<publication><port>8081</port><protocol>TCP</protocol><use_customer_network_address>1</use_customer_network_address></publication>"
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass -d '{"publication":{"port":"8080", "protocol":"TCP", "use_customer_network_address":"1"}}'
```

**Where:**

- **port** - the port for which this rule is active
- **protocol** - type of the protocol.
- **use_customer_network_address** - set 1 to use customer network address, otherwise set 0

**XML Output Example**

```xml
<publication>
  <created_at type="datetime">2012-09-14T13:17:41+03:00</created_at>
  <id type="integer">5</id>
  <is_built type="boolean">false</is_built>
  <outside_ip_address_id nil="true"></outside_ip_address_id>
  <port type="integer">8081</port>
  <protocol>TCP</protocol>
  <rule_number type="integer">55</rule_number>
  <updated_at type="datetime">2012-09-14T13:17:41+03:00</updated_at>
  <virtual_machine_id type="integer">66</virtual_machine_id>
</publication>
```

### 63.3 Delete Publishing Rule

To delete a publishing rule, use the following request:
DELETE /virtual_machines/:vm_id/publications/:id.xml
DELETE /virtual_machines/:vm_id/publications/:id.json

XML Request Example

curl -i -X DELETE -u user:userpass

JSON Request Example

curl -i -X DELETE -u user:userpass
64 Recipe Groups

Recipe groups allow OnApp administrators to organize individual recipes into groups that can be used as a bucket resource.

64.1 Get List of Recipe Groups

To view the list of recipe groups, use the following request:

GET /recipe_groups.xml
GET /recipe_groups.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<objects type="array">
  <object>
    <id type="integer">2</id>
    <label>qayd</label>
    <parent_id nil="true"/>
    <lft type="integer">11</lft>
    <rgt type="integer">12</rgt>
    <depth type="integer">0</depth>
    <created_at type="datetime">2013-05-24T11:50:19+03:00</created_at>
    <updated_at type="datetime">2013-05-24T11:50:19+03:00</updated_at>
  </object>
  <relations type="array">
    <relation>
      <id type="integer">1</id>
      <recipe_id type="integer">1</recipe_id>
      <recipe_group_id type="integer">2</recipe_group_id>
      <created_at type="datetime">2013-05-24T11:50:27+03:00</created_at>
      <updated_at type="datetime">2013-05-24T11:50:27+03:00</updated_at>
      <recipe>
        <id type="integer">1</id>
        <user_id type="integer">6</user_id>
        <created_at type="datetime">2013-05-24T11:50:27+03:00</created_at>
        <updated_at type="datetime">2013-05-24T11:50:27+03:00</updated_at>
        <label>Apache example</label>
        <description>Apache example</description>
        <script_type>bat</script_type>
        <compatible_with>unix</compatible_with>
      </recipe>
    </relation>
  </relations>
</object>
</objects>

Where:

objects - the array of recipe groups with the following parameters:

  label – the group name
  id - recipe group ID
  label - template group name
  parent_id - id of the target recipe group
  lft - left nested set identifier
  rgt - right nested set identifier
  depth - the depth of a given node (distance from this template group to the root)
  created_at – the date when the recipe group was created
  updated_at – the date when the recipe group was updated

children - the array of child recipe groups

relations - the array of recipes assigned to the recipe groups with the following parameters:

  id - relation ID
  user_id - ID of a recipe owner
  created_at – the date when the recipe group was created
  updated_at – the date when the recipe group was updated
• **recipe** - an array of recipe parameters:
  o **id** - recipe ID
  o **user_id** - ID of the ID of a recipe owner
  o **created_at** – the date when the recipe was created
  o **updated_at** – the date when the recipe was updated
  o **label** - recipe label
  o **script_type** - script type for Windows-compatible recipes:
    - bat
    - vbs
    - powershell
  o **compatible_with** - recipe compatibility: windows or unix

### 64.2 Get Recipe Group Details

To view the list of recipe groups, use the following request:

GET /recipe_groups/:recipe_group_id.xml
GET /recipe_groups/:recipe_group_id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where you have to specify the recipe group ID.

**XML Output Example**

```
<recipe_group>
  <created_at type="datetime">2013-05-24T08:54:13+00:00</created_at>
  <depth type="integer">1</depth>
  <id type="integer">6</id>
  <label>Decomissioning</label>
  <lft type="integer">8</lft>
  <parent_id type="integer">1</parent_id>
  <rgt type="integer">9</rgt>
  <updated_at type="datetime">2013-05-24T08:54:13+00:00</updated_at>
</recipe_group>
```

Where:

*label* – the group name
id - recipe group ID
label - template group name
parent_id - id of the target recipe group
lft - left nested set identifier
rgt - right nested set identifier
depth - the depth of a given node (distance from this template group to the root)
created_at – the date when the recipe group was created
updated_at – the date when the recipe group was updated

64.3 Add Recipe Group

To create a recipe group, use the following request:
POST /recipe_groups.xml
POST /recipe_groups.json

XML Request Example:
```
    curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
    '<recipe_group><label>test</label></recipe_group>' --url
    http://onapp.test/recipe_groups.xml
```

JSON Request Example:
```
    curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
    '{"recipe_group":{"label":"test"}}' --url http://onapp.test/recipe_groups.json
```

Where:
label - recipe group name

64.4 Add Child Recipe Group

To create a child recipe group, use the following request:
POST /recipe_groups.xml
POST /recipe_groups.json

XML Request Example:
```
    curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
    '<recipe_group><label>zaza</label><parent_id>100</parent_id></recipe_group>' --url
    http://onapp.test/recipe_groups.xml
```

JSON Request Example:
64.5 Edit Recipe Group

To edit recipe group's label, use the following request:

PUT /recipe_groups/:recipe_group_id.xml
PUT /recipe_groups/:recipe_group_id.json

**XML Request Example**

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<recipe_group><label>newlabel</label></recipe_group>' --url http://onapp.test/recipe_groups/13.xml
```

**JSON Request Example**

```
```

Where:

- **label** - the recipe group's label.

64.6 Delete Recipe Group

To delete a recipe group, use the following request:

DELETE /recipe_groups/:recipe_group_id.xml
DELETE /recipe_groups/:recipe_group_id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```
64.7 Get List of Recipes Attached to Recipe Group

To view the list of recipes attached to the recipe group, use the following request:

GET /recipe_groups/:id/recipe_group_relations.xml
GET /recipe_groups/:id/recipe_group_relations.json

**XML Request Example:**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/recipe_groups/2/recipe_group_relations.xml
```

**JSON Request Example:**

```
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/recipe_groups/2/recipe_group_relations.json
```

Where you have to specify ID of the recipe group in the URL.

64.8 Attach Recipe to Recipe Group

To attach recipe to the recipe group, use the following request:

POST /recipe_groups/:recipe_group_id/recipe_group_relations.xml
POST /recipe_groups/:recipe_group_id/recipe_group_relations.json

**XML Request Example**

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --data-urlencode '<recipe_group_relation><recipe_id>id</recipe_id></recipe_group_relation>' --url http://onapp.test/recipe_groups/13/recipe_group_relations.xml
```

**JSON Request Example**

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '
  "<recipe_group_relation><recipe_id>id</recipe_id></recipe_group_relation>"
' --url http://onapp.test/recipe_groups/13/recipe_group_relations.xml
```
64.9 Remove Recipe from Recipe Group

To remove recipe from the recipe group, use the following request:

**POST**

`/recipe_groups/:recipe_group_id/recipe_group_relations/:recipe_group_relation_id.xml`

**POST**

`/recipe_groups/:recipe_group_id/recipe_group_relations/:recipe_group_relation_id.json`

**XML Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/recipe_groups/2/recipe_group_relations/13.xml
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/recipe_groups/2/recipe_group_relations/13.json
```

Where you have to specify recipe group ID and ID of a recipe group relation in the URL.
65 Recipes

Recipes are the plugin mechanism used for adding new functionalities to the OnApp cloud. Each recipe is a set of instructions that triggers events at certain stages during the deployment of certain services.

In the 3.0.1 version you can use recipes for Unix (Linux and FreeBSD) virtual servers.

65.1 Get List of Recipes

To get the list of all recipes in your cloud, use the following request:

GET /recipes.xml
GET /recipes.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```xml
<recipes>
  <recipe>
    <name>Recipe 1</name>
    <description>First recipe</description>
    <action>runShellCommand</action>
    <stage>1</stage>
    <parameters/>
  </recipe>
  <recipe>
    <name>Recipe 2</name>
    <description>Second recipe</description>
    <action>installSoftware</action>
    <stage>2</stage>
    <parameters/>
  </recipe>
</recipes>
```
<recipes>
  <recipe>
    <compatible_with>unix</compatible_with>
    <created_at>2012-10-22T12:46:49+00:00</created_at>
    <description>Create initial folder structure</description>
    <id>1</id>
    <label>Create folders</label>
    <script_type>vbs</script_type>
    <updated_at>2012-10-22T12:46:49+00:00</updated_at>
    <user_id>2</user_id>
    <recipe_steps type="array">
      <recipe_step>
        <created_at>2012-10-22T12:46:49+00:00</created_at>
        <fail_anything_else>true</fail_anything_else>
        <fail_values/>
        <failure_goto_step>2</failure_goto_step>
        <id>1</id>
        <number>1</number>
        <on_failure>goto_step</on_failure>
        <on_success>proceed</on_success>
        <pass_anything_else>false</pass_anything_else>
        <pass_values>true</pass_values>
        <recipe_id>1</recipe_id>
        <result_source>exit_code</result_source>
        <script>mkdir /tmp</script>
        <success_goto_step>nil</success_goto_step>
        <updated_at>2012-10-22T12:46:49+00:00</updated_at>
      </recipe_step>
    </recipe_steps>
  </recipe>
  <recipe>
    <id>193</id>
    <user_id>8</user_id>
    <created_at>2019-03-04T11:03:48+02:00</created_at>
    <updated_at>2019-03-06T17:06:41+02:00</updated_at>
    <label>CP Revoke IP address</label>
    <description>Add ability to trigger recipes on add/remove IP address for a virtual server</description>
    <script_type>nil</script_type>
    <compatible_with>unix</compatible_with>
    <recipe_steps type="array">
      <recipe_step>
        <id>321</id>
        <recipe_id>193</recipe_id>
        <number>1</number>
        <script>
          echo "Add IP address. Step1" >> /tmp/recipe.txt; echo "$IPJOIN_IP_ADDRESS_ID $IPJOIN_IP_ADDRESS_ID" >> /tmp/IPJOIN_IP_ADDRESS_ID.txt; echo "IPJOIN_IP_ADDRESS $IPJOIN_IP_ADDRESS" >> /tmp/IPJOIN_IP_ADDRESS.txt; echo "IPJOIN_NETWORK_IDENTIFIER $IPJOIN_NETWORK_IDENTIFIER" >> /tmp/IPJOIN_NETWORK_IDENTIFIER.txt; echo "$IPJOIN_NIC_MAC_ADDRESS $IPJOIN_NIC_MAC_ADDRESS" >> /tmp/IPJOIN_NIC_MAC_ADDRESS.txt; echo "$IPJOIN_HYPERVISOR_IP_ADDRESS $IPJOIN_HYPERVISOR_IP_ADDRESS" >> /tmp/IPJOIN_HYPERVISOR_IP_ADDRESS.txt; echo "$TRIGGERING_EVENTS $TRIGGERING_EVENTS" >> /tmp/TRIGGERING_EVENTS.txt; echo "$IPJOIN_HYPERVISOR_NETWORK_INTERFACE $IPJOIN_HYPERVISOR_NETWORK_INTERFACE" >> /tmp/IPJOIN_HYPERVISOR_NETWORK_INTERFACE.txt;
        </script>
        <on_success>proceed</on_success>
        <on_failure>proceed</on_failure>
        <success_goto_step>nil</success_goto_step>
        <created_at>2019-03-04T11:04:17+02:00</created_at>
        <updated_at>2019-03-13T14:51:09+02:00</updated_at>
      </recipe_step>
    </recipe_steps>
  </recipe>
</recipes>
<result_source>exit_code</result_source>
<pass_values>0</pass_values>
<pass_anything_else type="boolean">false</pass_anything_else>
<fail_values/>
<fail_anything_else type="boolean">true</fail_anything_else>
<failure_goto_step nil="true"/>
</recipe_step>
</recipe_steps>
</recipe>
</recipes>

Where:

**compatible_with** - recipe compatibility: windows or unix

created_at — the date when the recipe was created in the [YYYY][MM][DD][hh][mm][ss]Z format

description - recipe description

id - recipe ID

label - recipe label

**script_type** - script type for Windows-compatible recipes:
- bat
- vbs
- powershell (PowerShell v1.0)

updated_at — the date when the recipe was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

use_on_hv_zones - true, if the recipe can be used on compute zones, otherwise false

use_on_vms - true, if the recipe can be used on virtual servers, otherwise false

user_id — the ID of a recipe owner

**recipe_steps** - an array of recipe steps with the following details:
- created_at — the date when the step was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- updated_at — the date when the step was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- fail_anything_else - set true, if you have specified the recipe pass value, otherwise set false

You can only specify behavior for one scenario: for example, if the fail_anything_else = false, passAnythingElse must be set to true.

- fail_values - recipe fail value
- failure_goto_step - if the on_failure parameter = goto_step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

- id - step ID
- number - step number
- script - recipe step code where you can specify the following values:
OnApp Cloud 6.3 API Guide

- **IPJOIN_IP_ADDRESS_ID** - ID of the IP Address
- **IPJOIN_IP_ADDRESS** - IP address as a string
- **IPJOIN_NETWORK_IDENTIFIER** - MAC address of network interface the IP address is assigned to
- **IPJOINNIC_MAC_ADDRESS** - MAC address of the network interface
- **IPJOIN_HYPERVERVISOR_IP_ADDRESS** - compute resource management IP address, the VS (with network interface with the assigned IP) is located on
- **TRIGGERING EVENTS** - events, which triggered the execution of a recipe
- **IPJOIN_HYPERVERVISOR_NETWORK_INTERFACE** - network interface on the compute resource to which the network is connected to

- **on_failure** - step behavior in case of failure:
  - **proceed** - proceed to the next step.
  - **fail** - terminate the recipe and mark it as failed.
  - **stop** - terminate the recipe and mark it as successful.
  - **go_to_step** - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

- **on_success** - step behavior in case of success:
  - **proceed** - proceed to the next step.
  - **fail** - terminate the recipe and mark it as failed.
  - **stop** - terminate the recipe and mark it as successful.
  - **go_to_step** - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

- **pass anything else** - set true, if you have specified the recipe fail value, otherwise set false
- **pass_values** - recipe pass value
- **recipe_id** - ID of a recipe the step belongs to
- **result_source** - step result source:
  - **exit_code** - an exit status, e.g. 0 will be returned on success
  - **std_out** - standard output
  - **std_err** - standard error
  - **std_out_and_std_err** - standard output and standard error

- **success goto step** - if the **on_success** parameter = goto step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

Page History

v.6.1
- added the **script** parameter

65.2 Get List of Servers Using Recipe

To get the list of servers that use a particular recipe, use the following request:

GET /recipes/recipe_id/applied_to_vs.xml
GET /recipes/recipe_id/applied_to_vs.json
XML Request Example

curl -i -X GET -u user:userpass http://onapp.test/recipes/12/applied_to_vs.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X GET -u user:userpass http://onapp.test/recipes/12/applied_to_vs.json -H 'Accept: application/json' -H 'Content-type: application/json'

XML Output Example

```xml
<virtual_machines type="array">
  <virtual_machine>
    <identifier>ha5iw2qlsu89vd</identifier>
    <template_id type="integer">594</template_id>
    <vm_events type="array">
      <vm_event>vm_provisioning</vm_event>
    </vm_events>
  </virtual_machine>
  <virtual_machine>
    <identifier>qxhokd7mgx1w69</identifier>
    <template_id type="integer">111</template_id>
    <vm_events type="array">
      <vm_event>vm_provisioning</vm_event>
    </vm_events>
  </virtual_machine>
</virtual_machines>
```

To assign recipes to multiple virtual or smart servers, see Run Recipe on Multiple Virtual Servers and Run Recipe on Multiple Smart Servers sections.

65.3 Get Recipe Details

To view recipe details, use the following request:

GET /recipes/:recipe_id.xml
GET /recipes/:recipe_id.json

XML Request Example


JSON Request Example

XML Output Example

```xml
<recipe>
  <compatible_with>unix</compatible_with>
  <created_at type="datetime">2012-10-22T12:46:49+00:00</created_at>
  <description>Create initial folder structure</description>
  <id type="integer">1</id>
  <script_type>vbs</script_type>
  <updated_at type="datetime">2012-10-22T12:46:49+00:00</updated_at>
  <user_id type="integer">2</user_id>
  <recipe_steps type="array">
    <recipe_step>
      <created_at type="datetime">2012-10-22T12:46:49+00:00</created_at>
      <fail_anything_else type="boolean">true</fail_anything_else>
      <fail_values/>
      <failure_goto_step type="integer">2</failure_goto_step>
      <id type="integer">1</id>
      <number type="integer">1</number>
      <on_failure>goto_step</on_failure>
      <on_success>proceed</on_success>
      <pass_anything_else type="boolean">false</pass_anything_else>
      <pass_values type="boolean">true</pass_values>
      <recipe_id type="integer">1</recipe_id>
      <result_source>exit_code</result_source>
      <script>mkdir /tmp</script>
      <success_goto_step nil="true"/>
      <updated_at type="datetime">2012-10-22T12:46:49+00:00</updated_at>
    </recipe_step>
  </recipe_steps>
</recipe>
```

Where:

- **compatible_with** - recipe compatibility: windows or unix

- **created_at** — the date when the recipe was created in the [YYYY][MM][DD][hh][mm][ss]Z format

- **description** - recipe description

- **id** - recipe ID

- **label** - recipe label

- **script_type** - script type for Windows-compatible recipes:
  - bat
  - vbs
  - powershell (PowerShell v1.0)

- **updated_at** — the date when the recipe was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

- **use_on_hv_zones** - true, if the recipe can be used on compute zones, otherwise false

- **use_on_vms** - true, if the recipe can be used on virtual servers, otherwise false

- **user_id** — the ID of a recipe owner

- **recipe_steps** - an array of recipe steps with the following details:
  - **created_at** — the date when the step was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** — the date when the step was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **fail_anything_else** - set true, if you have specified the recipe pass value, otherwise set false

You can only specify behavior for one scenario: for example, if the fail_anything_else = **false**, pass Anything_else must be set to **true**.

- **fail_values** - recipe fail value
- **failure_goto_step** - if the on_failure parameter = goto_step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.
- **id** - step ID
- **number** - step number
- **on_failure** - step behavior in case of failure:
  - **proceed** - proceed to the next step.
  - **fail** - terminate the recipe and mark it as failed.
  - **stop** - terminate the recipe and mark it as successful.
  - **go_to_step** - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.
- **on_success** - step behavior in case of success:
  - **proceed** - proceed to the next step.
  - **fail** - terminate the recipe and mark it as failed.
  - **stop** - terminate the recipe and mark it as successful.
  - **go_to_step** - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.
- **pass_anything_else** - set true, if you have specified the recipe fail value, otherwise set false
- **pass_values** - recipe pass value
- **recipe_id** - ID of a recipe the step belongs to
- **result_source** - step result source:
  - **exit_code** - an exit status, e.g. 0 will be returned on success
  - **std_out** - standard output
  - **std_err** - standard error
  - **std_out_and_std_err** - standard output and standard error
- **success_goto_step** - if the on_success parameter = goto_step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

Page History

v6.1
- added the **script** parameter
65.4 Add Recipe

To create a recipe, use the following request:

POST /recipes.xml
POST /recipes.json

**XML Request Example**

```bash
curl -i -X POST -u user:userpass http://onapp.test/recipes.xml -d
    '<recipe><label>New Label</label><description>New
description</description><compatible_with>windows</compatible_with><script
    _type>bat</script_type></recipe>' -H 'Accept: application/xml' -H
    'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass http://onapp.test/recipes.json -d
    '{"recipe":{"label":"New Label", "description":"New
Description","compatible_with":"windows","script_type":"bat"}}' -H
    'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **label** - recipe label
- **description** - recipe description
- **compatible_with** - recipe compatibility: windows or unix
- **script_type** - script type for Windows-compatible recipes:
  - bat
  - vbs
  - powershell (PowerShell v1.0)

This parameter is for Windows recipes only.

65.5 Edit Recipe

To edit a recipe, use the following request:

PUT /recipes/:recipe_id.xml
PUT /recipes/:recipe_id.json

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/recipes/12.xml -d
    '<recipe><label>New Label</label><description>New
description</description><compatible_with>windows</compatible_with><script
    _type>bat</script_type></recipe>' -H 'Accept: application/xml' -H
    'Content-type: application/xml'
```

**JSON Request Example**
Where:

- **label** - recipe label
- **description** - recipe description
- **compatible_with** - recipe compatibility: *windows* or *unix*
- **script_type** - script type for Windows-compatible recipes:
  - *bat*
  - *vbs*
  - *powershell* (PowerShell v1.0)

This parameter is for Windows recipes only.

### 65.6 Delete Recipe

To delete a recipe, use the following request:

DELETE /recipes/:recipe_id.xml
DELETE /recipes/:recipe_id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where:

- **recipe_id** - the recipe ID.

### 65.7 Manage Recipe Steps

Use the API calls provided in this section to create, edit and delete recipe steps.

#### 65.7.1 Get the List of Recipe Steps

To get the list of recipe steps, use the following request:
GET /recipes/:recipe_id/recipe_steps.xml
GET /recipes/:recipe_id/recipe_steps.json

**XML Request Example**

```bash
curl -i -X GET -u user:userpass
http://onapp.test/recipes/1/recipe_steps.xml -H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X GET -u user:userpass
http://onapp.test/recipes/1/recipe_steps.json -H 'Accept: application/json'
-H 'Content-type: application/json'
```

**XML Output Example**

```xml
textual_xml_output
```
<recipe>

<compatible_with>unix</compatible_with>
<created_at type="datetime">2013-04-19T10:26:41+03:00</created_at>
<description>Apache example</description>
{id type="integer">1</id>
<label>Apache example</label>
<script_type>bat</script_type>
<updated_at type="datetime">2013-05-13T13:08:00+03:00</updated_at>

<recipe_steps type="array">

<recipe_step>
<created_at type="datetime">2013-04-19T10:26:41+03:00</created_at>
<fail_anything_else type="boolean">true</fail_anything_else>
<fail_values></fail_values>
{id type="integer">1</id>
<number type="integer">1</number>
<on_failure>fail</on_failure>
<on_success>proceed</on_success>
<pass_anything_else type="boolean">false</pass_anything_else>
<pass_values></pass_values>
<recipe_id type="integer">1</recipe_id>
<result_source>exit_code</result_source>
<script>
# if echo $OPERATING_SYSTEM_DISTRO |grep rhel ; then
if rpm -qa |grep httpd |grep -v grep ; then
  yum -y update httpd
else
  yum -y install httpd
fi
# exit 1
</script>

<success_goto_step nil="true"></success_goto_step>
<updated_at type="datetime">2013-04-26T12:37:28+03:00</updated_at>
</recipe_step>

<recipe_step>
<created_at type="datetime">2013-04-23T13:54:02+03:00</created_at>
<fail_anything_else type="boolean">true</fail_anything_else>
<fail_values></fail_values>
<failure_goto_step type="integer">4</failure_goto_step>
{id type="integer">2</id>
<number type="integer">2</number>
<on_failure>goto_step</on_failure>
<on_success>goto_step</on_success>
<pass_anything_else type="boolean">false</pass_anything_else>
<pass_values></pass_values>
<recipe_id type="integer">1</recipe_id>
<result_source>exit_code</result_source>
<script>
  echo "&lt;p&gt;&lt;a href=http://$CP_ADDRESS&gt;OnApp Cloud&lt;/a&gt;&lt;/p&gt;" &gt; /var/www/html/index.html
</script>
<success_goto_step type="integer">5</success_goto_step>
<updated_at type="datetime">2013-04-26T13:03:26+03:00</updated_at>
</recipe_step>

<recipe_step>
<created_at type="datetime">2013-04-26T11:42:58+03:00</created_at>
<fail_anything_else type="boolean">true</fail_anything_else>
<fail_values></fail_values>
<failure_goto_step nil="true"></failure_goto_step>
{id type="integer">19</id>
<number type="integer">3</number>
<on_failure>fail</on_failure>
<on_success>stop</on_success>
<pass_anything_else type="boolean">false</pass_anything_else>
<pass_values></pass_values>
<recipe_id type="integer">1</recipe_id>
<result_source>exit_code</result_source>
</recipe_step>
</recipe>
Where:

compatible_with - recipe compatibility: windows or unix

created_at — the date when the recipe was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format

description - recipe description

id - recipe ID

label - recipe label

script_type - script type for Windows-compatible recipes:

- bat
- vbs
- powershell (PowerShell v1.0)
updated_at — the date when the recipe was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

use_on_hv_zones - true, if the recipe can be used on compute zones, otherwise false

use_on_vms - true, if the recipe can be used on virtual servers, otherwise false

user_id — the ID of a recipe owner

recipe_steps - an array of recipe steps with the following details:

- created_at — the date when the step was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- updated_at — the date when the step was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- fail Anything_else - set true, if you have specified the recipe pass value, otherwise set false

You can only specify behavior for one scenario: for example, if the fail Anything_else = false, pass Anything_else must be set to true.

- fail_values - recipe fail value
- failure_goto_step - if the on_failure parameter = goto_step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.
- id - step ID
- number - step number

- on_failure - step behavior in case of failure:
  - proceed - proceed to the next step.
  - fail - terminate the recipe and mark it as failed.
  - stop - terminate the recipe and mark it as successful.
  - go_to_step - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

- on_success - step behavior in case of success:
  - proceed - proceed to the next step.
  - fail - terminate the recipe and mark it as failed.
  - stop - terminate the recipe and mark it as successful.
  - go_to_step - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

- pass Anything_else - set true, if you have specified the recipe fail value, otherwise set false
- pass_values - recipe pass value
- recipe_id - ID of a recipe the step belongs to
- result_source - step result source:
  - exit_code - an exit status, e.g. 0 will be returned on success
  - std_out - standard output
  - std_err - standard error
  - std_out_and_std_err - standard output and standard error
• success_goto_step - if the on_success parameter = goto_step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

65.7.2 Create Recipe Step
To create a recipe step, use the following request:

POST /recipes/:recipe_id/recipe_steps.xml
POST /recipes/:recipe_id/recipe_steps.json

XML Request Example


JSON Request Example


Where:

script - recipe step code
result_source - step result source:
• exit_code - an exit status, e.g. 0 will be returned on success

To use exit code in the VBS or PowerShell scripts, you have to specify it directly in the script. For example:

**VBS**
Script:
WScript.Echo "test"
WScript.Quit 95

**PowerShell**
Script:
get-date -displayhint date
exit 227

• std_out - standard output
• std_err - standard error
• std_out_and_std_err - standard output and standard error
pass_anything_else - set true, if you have specified the recipe fail value, otherwise set false
pass_values - step pass value

on_success - step behavior in case of success:
  • proceed - proceed to the next step.
  • fail - terminate the recipe and mark it as failed.
  • stop - terminate the recipe and mark it as successful.
  • goto_step - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

success_goto_step - if the on_success parameter = goto_step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

fail_anything_else - set true, if you have specified the recipe pass value, otherwise set false
fail_values - step fail value

on_failure - step behavior in case of failure:
  • proceed - proceed to the next step.
  • fail - terminate the recipe and mark it as failed.
  • stop - terminate the recipe and mark it as successful.
  • goto_step - specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

failure_goto_step - if the on_failure parameter = goto_step, specify the step to proceed to. If you specify the nonexistent step, the recipe will be stopped.

You can only specify behavior for one scenario: for example, if the fail_anything_else = false, pass_anything_else must be set to true.

65.7.3 Edit Recipe Step

To edit a recipe step, use the following request:

PUT /recipes/:recipe_id/recipe_steps/:recipe_step_id.xml

PUT /recipes/:recipe_id/recipe_steps/:recipe_step_id.json

XML Request Example

```
curl -i -X PUT -u user:serpass
http://onapp.test/recipes/12/recipe_steps/32.xml  
'd.'<recipe_step><script>echo "123"</script></recipe_step>'  
-H 'Accept: application/xml'  
-H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X PUT -u user:userpass
http://onapp.test/recipes/12/recipe_steps/32.json  
'{"recipe_step":{"script":"echo abc"}}'  
-H 'Accept: application/json'  
-H 'Content-Type: application/json'
```

Where:
65.7.4 Remove Recipe Step
To remove a recipe step, use the following request:

PUT /recipes/:recipe_id/recipe_steps/:recipe_step_id.xml
PUT /recipes/:recipe_id/recipe_steps/:recipe_step_id.json

XML Request Example

```
curl -i -X DELETE -u user:userpass
    http://onapp.test/recipes/12/recipe_steps/21.xml
    -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X DELETE -u user:userpass
    http://onapp.test/recipes/12/recipe_steps/21.json
    -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:
In the URL, specify the ID of a recipe the step belongs to and the ID of a step you want to remove.

65.7.5 Swap Recipe Steps Locations
To interchange recipe steps, use the following request:

PUT http://onapp.test/recipes/:recipe_id/recipe_steps/:recipe_step_id/move_to/:recipe_step_number.xml
PUT http://onapp.test/recipes/:recipe_id/recipe_steps/:recipe_step_id/move_to/:recipe_step_number.json

XML Request Example

```
curl -i -X PUT -u user:userpass
    http://onapp.test/recipes/12/recipe_steps/21/move_to/31.xml
    -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X PUT -u user:userpass
    http://onapp.test/recipes/12/recipe_steps/21/move_to/31.json
    -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where you need to specify the IDs of recipes you want to move in the URL.

65.8 Manage Virtual Server Recipes
Use the following API calls to view, assign and delete virtual server recipes in your cloud.
65.8.1 Get List of Virtual Server Recipes

To get the list of virtual server recipes, use the following request:

GET /virtual_machines/:virtual_machine_id/recipe_joins.xml
GET /virtual_machines/:virtual_machine_id/recipe_joins.json

XML Request Example

```bash
curl -i -X GET -u user:userpass
http://onapp.test/virtual_machines/13/recipe_joins.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass
http://onapp.test/virtual_machines/13/recipe_joins.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Shows the same recipe attributes as in the Get List of Recipes section.

65.8.2 Assign Recipe to Virtual Server

To assign a recipe to a VS, use the following request:

POST /virtual_machines/:virtual_machine_id/recipe_joins.xml
POST /virtual_machines/:virtual_machine_id/recipe_joins.json

XML Request Example

```bash
curl -i -X POST -H "Accept: application/xml" -H "Content-type: application/xml" -d
"<recipe_join><recipe_id>6</recipe_id><event_type>vm_provisioning</event_type></recipe_join>"
-u user:userpass
http://onapp.test/virtual_machines/13/recipe_joins.xml
```

JSON Request Example

```bash
curl -i -X POST -H "Accept: application/json" -H "Content-type: application/json" -d
'{"recipe_join":{"recipe_id":"6","event_type":"vm_provisioning"}}' -u user:userpass
http://onapp.test/virtual_machines/13/recipe_joins.json
```

Where:

- **recipe_join** - an array of recipe join details:
  - **recipe_id** - ID of a recipe you want to assign
  - **event_type** - type of the event you want to assign the recipe to:
    - **vm_provisioning** - run the recipe during VS provisioning
    - **vm_network_rebuild** - run the recipe when rebuilding a network
    - **vm_disk_add** - run the recipe when adding a disk
OnApp Cloud 6.3 API Guide

- **ip_allocated_to_vm_nic** - run the recipe when adding an IP address to the VS network interface
- **ip_revoked_from_vm_nic** - run the recipe when removing an IP address from the VS network interface
- **vm_nic_add** - run the recipe when adding a network interface
- **vm_nic_remove** - run the recipe while deleting a network interface
- **vm_disk_resize** - run the recipe when resizing a VS disk
- **vm_resize** - run the recipe when resizing a VS
- **vm_ip_address_add** - run the recipe when adding an IP address to a VS
- **vm_ip_address_remove** - run the recipe when removing an IP address from a VS
- **vm_start** - run the recipe while starting the virtual server
- **vm_reboot** - run the recipe while rebooting the virtual server
- **vm_hot_migrate** - run the recipe during the hot migration of the virtual server
- **vm_hot_full_migrate** - run the recipe during the hot migration of the virtual server with disk
- **vm_failover** - run the recipe during the failover process

- **virtual_machine_id** - ID of a virtual server you want to assign the recipe to

**Page History**

v. 6.1

- added the following parameters:
  - **ip_allocated_to_vm_nic**
  - **ip_revoked_from_vm_nic**
  - **vm_nic_remove**
  - **vm_ip_address_add**
  - **vm_ip_address_remove**
  - **vm_start**
  - **vm_reboot**
  - **vm_hot_migrate**
  - **vm_hot_full_migrate**
  - **vm_failover**

### 65.8.3 Remove Recipe from Virtual Server

To remove a recipe from a virtual server, use the following request:

```
DELETE
/virtual_machines/:virtual_machine_id/recipe_joins/:recipe_join_id.xml
```
DELETE
/virtual_machines/:virtual_machine_id/recipe_joins/:recipe_join_id.json

XML Request Example

```
curl -i -X DELETE -H "Accept: application/xml" -H "Content-type: application/xml" -u user:userpass
  http://onapp.test/virtual_machines/h76rawywphxk6/recipe_joins/1.xml
```

JSON Request Example

```
curl -i -X DELETE -H "Accept: application/json" -H "Content-type: application/json" -u user:userpass
  http://onapp.test/virtual_machines/h76rawywphxk6/recipe_joins/1.json
```

In the URL, specify the ID of a virtual server and the ID of a recipe you want to remove.

### 65.8.4 Run Recipe on Multiple Virtual Servers

To run a recipe on multiple virtual servers, use the following request:

**POST /recipes/:recipe_id/run.xml**

**POST /recipes/:recipe_id/run.json**

**XML Request Example**

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<virtual_machines type="array"><virtual_machine>vm1_identifier</virtual_machine><virtual_machine>vm2_identifier</virtual_machine></virtual_machines>' --url
  http://onapp.test/recipes/13/run.xml
```

**JSON Request Example**

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:password -d '{"virtual_machines":[":vm1_identifier",":vm2_identifier"]}' --url
  http://onapp/recipes/13/run.json
```

Where:

- **vm1_identifier** - replace this with the identifier of a virtual server you want to run the recipe on.
- ID of the required recipe in the URL.

This request returns the multi status code - 207.

When applying one recipe to several servers via API, there is possibility to run the recipe incompatible with the server type (Unix recipe on Windows...
server or vice versa). In this case the transaction will be scheduled and completed, but the recipe will not do anything.

To view the list of servers the particular recipe is applied to, see the Get List of Servers Using Recipe request.

65.9 Manage Smart Server Recipes

Use the following API calls to view, assign and delete smart server recipes in your cloud.

65.9.1 Get List of Smart Server Recipes

To get the list of smart server recipes, use the following request:

GET /smart_servers/:smart_server_id/recipe_joins.xml
GET /smart_servers/:smart_server_id/recipe_joins.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

Shows the same recipe attributes as in the Get List of Recipes section.

65.9.2 Assign Recipe to Smart Server

To assign recipe to a smart server, use the following request:

POST /smart_servers/:smart_server_id/recipe_joins.xml
POST /smart_servers/:smart_server_id/recipe_joins.json

XML Request Example

```bash
```

JSON Request Example
curl -i -X POST -H "Accept: application/json" -H "Content-Type: application/json" -d '{"recipe_join":{"recipe_id":"6", "event_type":"vm_provisioning"}, "smart_server_id":"h76rawywpkhx6"}' -u user:userpass
http://onapp.test/smart_servers/h76rawywpkhx6/recipe_joins.json

Where:

recipe_join - an array of recipe join details:

- recipe_id - ID of a recipe you want to assign
- event_type - type of the event you want to assign the recipe to:
  - vm_provisioning - run the recipe during VS provisioning
  - vm_network_rebuild - run the recipe when rebuilding a network
  - vm_disk_add - run the recipe when adding a disk
  - vm_nic_add - run the recipe when adding a network interface
- ip_allocated_to_vm_nic - run the recipe when adding an IP address to the VS network interface
- ip_revoked_from_vm_nic - run the recipe when removing an IP address from the VS network interface
- vm_nic_remove - run the recipe while deleting a network interface
- vm_disk_resize - run the recipe when resizing a VS disk
- vm_resize - run the recipe when resizing a VS
- vm_ip_address_add - run the recipe when adding an IP address to a VS
- vm_ip_address_remove - run the recipe when removing an IP address from a VS
- vm_start - run the recipe while starting the virtual server
- vm_reboot - run the recipe while rebooting the virtual server
- vm_hot_migrate - run the recipe during the hot migration of the virtual server
- vm_hot_full_migrate - run the recipe during the hot migration of the virtual server with disk
- vm_failover - run the recipe during the failover process
- smart_server_id - ID of a smart server you want to assign the recipe to

Page History

v. 6.1

- added the following parameters:
  - ip_allocated_to_vm_nic
  - ip_revoked_from_vm_nic
  - vm_nic_remove
  - vm_ip_address_add
65.9.3 Remove Recipe from Smart Server

To remove a recipe from a smart server, use the following request:

DELETE
/smart_servers/:smart_server_id/recipe_joins/:recipe_join_id.xml
DELETE
/smart_servers/:smart_server_id/recipe_joins/:recipe_join_id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

In the URL, specify the ID of a smart server and the ID of a recipe you want to remove.

65.9.4 Run Recipe on Multiple Smart Servers

To run a recipe on multiple smart servers, use the following request:

POST /recipes/:recipe_id/run.xml
POST /recipes/:recipe_id/run.json

**XML Request Example**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<?xml version="1.0" encoding="utf-8"?>
<smart_servers type="array"><smart_server>identifier1</smart_server><smart_server>identifier2</smart_server></smart_servers>' --url http://onapp.test/recipes/12/run.xml
```

**JSON Request Example**

```bash
```
Where:

**identifier** - identifier of a smart server you want to run the recipe on

ID of the required recipe in the URL

When applying one recipe to several servers via API, there is possibility to run the recipe incompatible with the server type (Unix recipe on Windows server or vice versa). In this case the transaction will be scheduled and completed, but the recipe will not do anything.

To view the list of server the particular recipe is applied to, see the **Get List of Servers Using Recipe** request.

### 65.10 Manage Baremetal Server Recipes

Use the following API calls to view, assign and delete baremetal server recipes in your cloud.

#### 65.10.1 Get the List of Baremetal Server Recipes

To get the list of baremetal server recipes, use the following request:

```
GET /baremetal_servers/:baremetal_server_id/recipe_joins.xml
GET /baremetal_servers/:baremetal_server_id/recipe_joins.json
```

**XML Request Example**

```
curl -i -X GET -u user:userpass
    http://onapp.test/baremetal_servers/12/recipe_joins.xml
    -H 'Accept: application/xml'
    -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X GET -u user:userpass
    http://onapp.test/baremetal_servers/12/recipe_joins.json
    -H 'Accept: application/json'
    -H 'Content-type: application/json'
```

Shows the same recipe attributes as in the **Get List of Recipes** section.

#### 65.10.2 Assign Recipe to Baremetal Server

You can only assign recipes to baremetal server during the server creation process. Refer to the **Create Baremetal Server** section for details.

#### 65.10.3 Remove Recipe from Baremetal Server

To remove a recipe from a baremetal server, use the following request:

```
DELETE /baremetal_servers/:baremetal_server_id/recipe_joins/:recipe_join_id.xml
DELETE
```
/baremetal_servers/:baremetal_server_id/recipe_joins/:recipe_join_id.j
gson

**XML Request Example**

curl -i -X DELETE -H "Accept: application/xml" -H "Content-
type:application/xml" -u user:userpass
http://onapp.test/baremetal_servers/h76rawyvwphxk6/recipe_joins/6.xml

**JSON Request Example**

curl -i -X DELETE -H "Accept: application/json" -H "Content-
type:application/json" -u user:userpass
http://onapp.test/baremetal_servers/h76rawyvwphxk6/recipe_joins/6.json

In the URL, specify the ID of a baremetal server and the ID of a recipe you want to remove.

### 65.11 Manage Template Recipes

Use the following API calls to view, assign and delete template recipes in your cloud.

#### 65.11.1 Get the List of Template Recipes

To get the list of template recipes, use the following request:

GET /templates/:template_id/recipe_joins.xml
GET /templates/:template_id/recipe_joins.json

**XML Request Example**

curl -i -X GET -u user:userpass

**JSON Request Example**

curl -i -X GET -u user:userpass

Shows the same recipe attributes as in the Get List of Recipes section.

#### 65.11.2 Assign Recipe to Template

To assign a recipe to a template, use the following request:

POST /templates/:template_id/recipe_joins.xml
POST /templates/:template_id/recipe_joins.json

**XML Request Example**

curl -i -X POST -u user:userpass

**JSON Request Example**

curl -i -X POST -u user:userpass
curl -i -X POST -H "Accept: application/xml" -H "Content-type:application/xml" -d
'<?xml version="1.0" encoding="UTF-8"?>
<recipe_join><recipe_id>6</recipe_id><event_type>vm_provisioning</event_type></recipe_join>' -u user:userpass
http://onapp.test/templates/6/recipe_joins.xml

JSON Request Example

curl -i -X POST -H "Accept: application/json" -H "Content-type:application/json" -d
'{"recipe_join":{"recipe_id":"6","event_type":"vm_provisioning"},"template_id":"1"}' -u user:userpass
http://onapp.test/templates/6/recipe_joins.json

Where:

- **recipe_id** - ID of a recipe you want to assign
- **event_type** - type of the event you want to assign the recipe to:
  - *vm_provisioning* - run the recipe during VS provisioning
  - *vm_network_rebuild* - run the recipe when rebuilding a network
  - *vm_disk_add* - run the recipe when adding a disk
  - *ip_allocated_to_vm_nic* - run the recipe when adding an IP address to the VS network interface
  - *ip_revoked_from_vm_nic* - run the recipe when removing an IP address from the VS network interface
  - *vm_nic_add* - run the recipe when adding a network interface
  - *vm_nic_remove* - run the recipe while deleting a network interface
  - *vm_disk_resize* - run the recipe when resizing a VS disk
  - *vm_resize* - run the recipe when resizing a VS
  - *vm_ip_address_add* - run the recipe when adding an IP address to a VS
  - *vm_ip_address_remove* - run the recipe when removing an IP address from a VS
  - *vm_start* - run the recipe while starting the virtual server
  - *vm_reboot* - run the recipe while rebooting the virtual server
  - *vm_hot_migrate* - run the recipe during the hot migration of the virtual server
  - *vm_hot_full_migrate* - run the recipe during the hot migration of the virtual server with disk
  - *vm_failover* - run the recipe during the failover process
- **template_id** - ID of a template you want to assign the recipe to

Page History

v. 6.1
- added the following parameters:
65.11.3 Remove Recipe from Template

To remove a recipe from a template, use the following request:

DELETE /templates/:template_id/recipe_joins/:recipe_join_id.xml
DELETE /templates/:template_id/recipe_joins/:recipe_join_id.json

**XML Request Example**

```
curl -i -X DELETE -H "Accept: application/xml"  -H "Content-type:application/xml" -u user:userpass
http://onapp.test/templates/2/recipe_joins/9.xml
```

**JSON Request Example**

```
curl -i -X DELETE -H "Accept: application/json"  -H "Content-type:application/json" -u user:userpass
http://onapp.test/templates/2/recipe_joins/9.json
```

In the URL, specify the template ID and the ID of a recipe you want to remove.

65.12 Manage Compute Zone Recipes

Use the following API calls to view, assign and delete compute zone recipes in your cloud.

65.12.1 Get the List of Compute Zone Recipes

To get the list of compute zone recipes, use the following request:

GET /settings/hypervisor_zones/:hypervisor_zone_id/recipe_joins.xml
GET /settings/hypervisor_zones/:hypervisor_zone_id/recipe_joins.json

**XML Request Example**

```
curl -i -X GET -H "Accept: application/xml"  -H "Content-type:application/xml" -u user:userpass
http://onapp.test/settings/hypervisor_zones/2/recipe_joins.xml
```

**JSON Request Example**

```
curl -i -X GET -H "Accept: application/json"  -H "Content-type:application/json" -u user:userpass
http://onapp.test/settings/hypervisor_zones/2/recipe_joins.json
```
curl -i -X GET -u user:userpass  

JSON Request Example

curl -i -X GET -u user:userpass  

Shows the same recipe attributes as in the Get List of Recipes section.

### 65.12.2 Assign Recipe to Compute Zone

To assign a recipe to a compute zone, use the following request:

**XML Request Example**

```bash
curl -i -X POST -H "Accept: application/xml" -H "Content-type:application/xml" -d "<recipe_join><recipe_id>6</recipe_id><event_type>vm_provisioning</event_type><template_id>h76rawyvwphxk6</template_id></recipe_join>" -u user:userpass  
http://onapp.test/settings/hypervisor_zones/9/recipe_joins.xml
```

**JSON Request Example**

```bash
curl -i -X POST -H "Accept: application/json" -H "Content-type:application/json" -d '{"recipe_join":{"recipe_id":"6", "event_type":"vm_provisioning"}, "template_id":"h76rawyvwphxk6"}' -u user:userpass  
http://onapp.test/settings/hypervisor_zones/9/recipe_joins.json
```

*Where:*

- **recipe_id** - ID of a recipe you want to assign
- **event_type** - type of the event you want to assign the recipe to:
  - `hv_goes_online` - run the recipe when the compute resource comes online
  - `hv_goes_offline` - run the recipe when the compute resource goes offline
  - `vm_provisioning` - run the recipe during VS provisioning
  - `vm_network_rebuild` - run the recipe when rebuilding a network
  - `vm_disk_add` - run the recipe when adding a disk

**NOTE:** The recipe will be triggered when the statistics is not received from a compute resource for a certain period of time for some reason. If the compute resource is offline, the recipe will not run.
- `ip_allocated_to_vm_nic` - run the recipe when adding an IP address to the VS network interface
- `ip_revoked_from_vm_nic` - run the recipe when removing an IP address from the VS network interface
- `vm_nic_add` - run the recipe when adding a network interface
- `vm_nic_remove` - run the recipe while deleting a network interface
- `vm_disk_resize` - run the recipe when resizing a VSs disk
- `vm_resize` - run the recipe when resizing a VS
- `vm_ip_address_add` - run the recipe when adding an IP address to a VS
- `vm_ip_address_remove` - run the recipe when removing an IP address from a VS
- `vm_start` - run the recipe while starting the virtual server
- `vm_reboot` - run the recipe while rebooting the virtual server
- `vm_hot_migrate` - run the recipe during the hot migration of the virtual server
- `vm_hot_full_migrate` - run the recipe during the hot migration of the virtual server with disk
- `vm_failover` - run the recipe during the failover process

`hypervisor_zone_id` - ID of a compute zone you want to assign the recipe to

**Page History**

v. 6.1
- added the following parameters:
  - `ip_allocated_to_vm_nic`
  - `ip_revoked_from_vm_nic`
  - `vm_nic_remove`
  - `vm_ip_address_add`
  - `vm_ip_address_remove`
  - `vm_start`
  - `vm_reboot`
  - `vm_hot_migrate`
  - `vm_hot_full_migrate`
  - `vm_failover`

**65.12.3 Remove Recipe from Compute Zone**

To remove a recipe from a compute zone, use the following request:

```plaintext
delete /settings/hypervisor_zones/:hypervisor_zone_id/recipe_joins/:recipe_join_id.xml
```

DELETE
Get the List of Control Panel Recipes
To get the list of control panel recipes, use the following request:

GET /settings/control_panel/recipe_joins.xml
GET /settings/control_panel/recipe_joins.json

XML Request Example

```bash
curl -i -X GET -u user:userpass
http://onapp.test/settings/control_panel/recipe_joins.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml'
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass
http://onapp.test/settings/control_panel/recipe_joins.json -H 'Accept: application/json' -H 'Content-Type: application/json'
```

Shows the same recipe attributes as in the Get List of Recipes section.

Assign Recipe to Control Panel
To assign a recipe to a Control Panel, use the following request:

POST /settings/control_panel/recipe_joins.xml
POST /settings/control_panel/recipe_joins.json

XML Request Example

```bash
curl -i -X DELETE -H "Accept: application/xml" -H "Content-Type:application/xml" -u user:userpass
```
curl -i -X POST -H "Accept: application/xml" -H "Content-type:application/xml" -d
' <recipe_join><recipe_id>6</recipe_id><event_type>vm_provisioning</event_type></recipe_join>' -u user:userpass
http://onapp.test/settings/control_panel/recipe_joins.xml

JSON Request Example

curl -i -X POST -H "Accept: application/json" -H "Content-type:application/json" -d
'{"recipe_join":{"recipe_id":"6","event_type":"vm_provisioning"}}' -u user:userpass
http://onapp.test/settings/control_panel/recipe_joins.json

Where:

**recipe_id** - ID of a recipe you want to assign

**event_type** - type of the event you want to assign the recipe to:

- **kvm_hv_goes_online** - run the recipe when the KVM compute resource comes online
- **kvm_hv_goes_offline** - run the recipe when the KVM compute resource goes offline
- **xen_hv_goes_online** - run the recipe when the Xen compute resource comes online
- **xen_hv_goes_offline** - run the recipe when the Xen compute resource goes offline
- **vmware_hv_goes_online** - run the recipe when the VMware compute resource comes online
- **vmware_hv_goes_offline** - run the recipe when the VMware compute resource goes offline
- **hv_added** - run the recipe when the new compute resource is added
- **hv_removed** - run the recipe when compute resource is removed
- **ip_allocated_to_vm_nic** - run the recipe when adding an IP address to the VS network interface
- **ip_revoked_from_vm_nic** - run the recipe when removing an IP address from the VS network interface
- **vm_provisioning** - run the recipe during VS provisioning
- **vm_network_rebuild** - run the recipe when rebuilding a network
- **vm_disk_add** - run the recipe when adding a disk
- **vm_nic_add** - run the recipe when adding a network interface
- **vm_nic_remove** - run the recipe while deleting a network interface
- **vm_disk_resize** - run the recipe when resizing a VS disk
- **vm_resize** - run the recipe when resizing a VS
- **vm_ip_address_add** - run the recipe when adding an IP address to a VS
- **vm_ip_address_remove** - run the recipe when removing an IP address from a VS

**NOTE:** The recipe will be triggered when the statistics is not received from a compute resource for a certain period of time for some reason. If the compute resource is offline, the recipe will not run.
- **vm_start** - run the recipe while starting the virtual server
- **vm_reboot** - run the recipe while rebooting the virtual server
- **vm_hot_migrate** - run the recipe during the hot migration of the virtual server
- **vm_hot_full_migrate** - run the recipe during the hot migration of the virtual server with disk
- **vm_failover** - run the recipe during the failover process

Page History
v. 6.1
- added the following parameters:
  - **ip_allocated_to_vm_nic**
  - **ip_revoked_from_vm_nic**
  - **vm_nic_remove**
  - **vm_ip_address_add**
  - **vm_ip_address_remove**
  - **vm_start**
  - **vm_reboot**
  - **vm_hot_migrate**
  - **vm_hot_full_migrate**
  - **vm_failover**

### 65.13.3 Remove Recipe from Control Panel

To remove a recipe from the Control Panel, use the following request:

DELETE settings/control_panel/recipe_joins/:recipe_id.xml
DELETE settings/control_panel/recipe_joins/:recipe_id.json

**XML Request Example**

```
curl -i -X DELETE -H "Accept: application/xml" -H "Content-Type: application/xml" -u user:userpass
http://onapp.test/settings/control_panel/recipe_joins/9.xml
```

**JSON Request Example**

```
curl -i -X DELETE -H "Accept: application/json" -H "Content-Type: application/json" -u user:userpass
http://onapp.test/settings/control_panel/recipe_joins/9.json
```

In the URL, specify the ID of a recipe you want to remove.
66 Resolvers

Resolvers translate hostnames to IP addresses. At least two resolvers should be specified for each network in the system. View, edit and delete commands are available for existing resolvers.
Resolvers are known as name servers in the API.

66.1 Get List of Resolvers

To get the list of all available resolvers in your cloud, use the following request:

GET /settings/nameservers.xml
GET /settings/nameservers.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<nameservers type="array">
<nameserver>
<address>8.8.8.8</address>
<created_at type="datetime">2011-02-14T15:55:44+02:00</created_at>
<network_id type="integer">1</network_id>
<updated_at type="datetime">2011-02-14T15:55:44+02:00</updated_at>
{id type="integer">1</id>
...</nameserver>
...</nameservers>
```

Where:

- **address** - the resolver IP address
- **created_at** - the timestamp in database when this record was created
- **network_id** - the ID of the network to which this resolver belongs
- **updated_at** - the timestamp in database to which this resolver belongs
- **id** - the ID of this resolver

66.2 Get Resolver Details

To get details for a particular resolver, use the following request:
GET /settings/nameservers/:id.xml
GET /settings/nameservers/:id.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```
<nameserver>
  <address>8.8.8.8</address>
  <created_at type="datetime">2011-02-14T15:55:44+02:00</created_at>
  <network_id type="integer">1</network_id>
  <updated_at type="datetime">2011-02-14T15:55:44+02:00</updated_at>
  <id type="integer">1</id>
</nameserver>
```

The parameters are the same as for Get List of Resolvers request.

### 66.3 Add Resolver

To add a new resolver, use the following request:

POST /settings/nameservers.xml
POST /settings/nameservers.json

**XML Request Example**


**JSON Request example**


**Where:**

- `address`* - the resolver IP address
- `network_id`* - the ID of the network to which this resolver should belong
66.4 Edit Resolver

To edit a resolver, use the following request:

PUT /settings/nameservers/:id.xml
PUT /settings/nameservers/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:

- **address** - the resolver IP address
- **network_id** - the ID of the network to which this resolver should belong

66.5 Delete Resolver

To delete a resolver, use the following request:

DELETE /settings/nameservers/:id.xml
DELETE /settings/nameservers/:id.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/nameservers/12.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/settings/nameservers/12.json
```

Returns HTTP 200 response on successful deletion, or HTTP 404 when a resolver with the ID specified is not found, or the URL requested is incorrect.
67 Restrictions Sets

Restrictions set is a customizable group of limitations. This tool gives cloud administrators more flexibility in limiting resources and operations available to reseller role(s). Creating a new restrictions set associates a role or number of roles with certain resources’ limitations.

This section lists the API calls required for configuring restrictions sets.

67.1 Get List of Restrictions Sets

To see all restrictions sets, use the following request:

GET /restrictions/sets.xml
GET /restrictions/sets.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**
Where:

**restrictions_sets** - the array of parameters associated with the restrictions set(s):

- **created_at** - the date when the restrictions set was created
- **id** - ID of the restrictions set
- **identifier** - identifier of the restrictions set in the DB
- **label** - restrictions set name
- **updated_at** - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

**roles** - the array of parameters associated with role(s) assigned to the restrictions set:

- **created_at** - the date when the role was created
- **id** - ID of the role
- **identifier** - identifier of the role in the DB
- **label** - role name
- **updated_at** - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- **users_count** - the number of users assigned to the role

**restrictions_resources** - the array of parameters associated with the restrictions resource(s) limited by the restrictions set:

- **created_at** - the date when the restrictions resource was created
- **id** - ID of the restrictions resource
- **identifier** - identifier of the restrictions resource in the DB
- **restriction_type** - the type of restriction - either by user group or bucket
updated_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
label - the restrictions resource name

67.2 Get Restrictions Set Details

To get the details of particular restrictions set, use the following request:
GET /restrictions/sets/:id.xml
GET /restrictions/sets/:id.json

XML Request Example

```
curl -i -X GET http://onapp.test/restrictions/sets/7.xml -u user:userpass
```

XML Request Example

```
curl -i -X GET http://onapp.test/restrictions/sets/7.json -u user:userpass
```

XML Output Example

```
<restrictions_set>
  <created_at type="datetime">2015-01-26T15:32:23+02:00</created_at>
  <id type="integer">7</id>
  <identifier>v9dwt9f95h00k</identifier>
  <label>label</label>
  <updated_at type="datetime">2015-01-26T15:32:23+02:00</updated_at>
  <roles type="array">
    <role>
      <created_at type="datetime">2015-01-26T15:29:39+02:00</created_at>
      <id type="integer">17</id>
      <identifier>hlj7q4vnxczezt</identifier>
      <label>marta.test.role</label>
      <updated_at type="datetime">2015-01-29T15:36:24+02:00</updated_at>
      <users_count type="integer">2</users_count>
    </role>
  </roles>
  <restrictions_resources type="array">
    <restrictions_resource>
      <created_at type="datetime">2014-12-25T12:50:34+02:00</created_at>
      <id type="integer">2</id>
      <identifier>auto_scaling_configurations</identifier>
      <restriction_type>by_user_group</restriction_type>
      <updated_at type="datetime">2014-12-25T12:50:34+02:00</updated_at>
      <label>Autoscaling Configuration (by User Group)</label>
    </restrictions_resource>
  </restrictions_resources>
</restrictions_set>
```

Where:

restrictions_set - the array of parameters associated with the restrictions set:

  created_at - the date when the restrictions set was created
  id - ID of the restrictions set
  identifier - identifier of the restrictions set in the DB
label - restrictions set name

updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

roles - the array of parameters associated with role(s) assigned to the restrictions set:

  created_at - the date when the role was created
  id - ID of the role
  identifier - identifier of the role in the DB
  label - role name
  updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
  users_count - the number of users assigned to the role

restrictions_resources - the array of parameters associated with the restrictions resource(s) limited by the restrictions set:

  created_at - the date when the restrictions resource was created
  id - ID of the restrictions resource
  identifier - identifier of the restrictions resource in the DB
  restriction_type - the type of restriction - either by user group or bucket
  updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
  label - the restrictions resource name

67.3 Get List of All Restrictions Resources

To see all restrictions resources, use the following request:

GET /restrictions/resources.xml
GET /restrictions/resources.json

XML Request Example

curl -i -X GET http://onapp.test/restrictions/resources.xml -u user:userpass

JSON Request Example

curl -i -X GET http://onapp.test/restrictions/resources.json -u user:userpass

XML Output Example
<restrictions_resources_type="array">
<restrictions_resource>
    <created_at type="datetime">2014-12-25T12:50:34+02:00</created_at>
    <id type="integer">1</id>
    <identifier>activity_logs</identifier>
    <restriction_type>by_user_group</restriction_type>
    <updated_at type="datetime">2014-12-25T12:50:34+02:00</updated_at>
    <label>Activity Logs (by User Group)</label>
</restrictions_resource>
<restrictions_resource>
    <created_at type="datetime">2014-12-25T12:50:34+02:00</created_at>
    <id type="integer">2</id>
    <identifier>auto_scaling_configurations</identifier>
    <restriction_type>by_user_group</restriction_type>
    <updated_at type="datetime">2014-12-25T12:50:34+02:00</updated_at>
    <label>Autoscaling Configuration (by User Group)</label>
</restrictions_resource>
<restrictions_resource>
    <created_at type="datetime">2014-12-25T12:50:34+02:00</created_at>
    <id type="integer">3</id>
    <identifier>auto_scaling_configurations</identifier>
    <restriction_type>by_billing_plan</restriction_type>
    <updated_at type="datetime">2014-12-25T12:50:34+02:00</updated_at>
    <label>Autoscaling Configuration (by Billing Plan)</label>
</restrictions_resource>
<restrictions_resource>
    <created_at type="datetime">2014-12-25T12:50:34+02:00</created_at>
    <id type="integer">4</id>
    <identifier>backup_server_groups</identifier>
    <restriction_type>by_billing_plan</restriction_type>
    <updated_at type="datetime">2014-12-25T12:50:34+02:00</updated_at>
    <label>Backup Server Zones (by Billing Plan)</label>
</restrictions_resource>
</restrictions_resources>

Where:

restrictions_resources - the array of parameters associated with the restrictions resource(s) limited by the restrictions set:

created_at - the date when the record in the database was created
id - ID of the restrictions resource
identifier - identifier of the restrictions resource in the DB
restriction_type - the type of restriction - either by user group or bucket
updated_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
label - the restrictions resource name

67.4 Create Restrictions Set

To create a restrictions set, use the following request:
POST /restrictions/sets.xml
POST /restrictions/sets.json

XML Request Example
curl -i -X POST http://onapp.test/restrictions/sets.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<restrictions_set><label>label</label><role_ids type="array"><role_id>1</role_id></role_ids><resource_ids type="array"><resource_id>1</resource_id></resource_ids></restrictions_set>'

**JSON Request Example**

curl -i -X POST http://onapp.test/restrictions/sets.json -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"restrictions_set":{"label":"label","role_ids":[1],"resource_ids":[1]}}'

Where:

* label - give the label of the restrictions set
* role_ids - the array of the role IDs you wish to assign to the restrictions set:
  * role_id - ID of the role you wish to assign to the restrictions set
* resource_ids - the array of resource IDs you wish to limit under the restrictions set:
  * resource_id - ID of the resource you wish to limit under the restrictions set

### 67.5 Edit Restrictions Set

To edit a restrictions set, use the following request:

PUT /restrictions/sets/:id.xml

PUT /restrictions/sets/:id.json

**XML Request Example**


**JSON Request Example**


Where:

* label - give the label of the restrictions set
* role_ids - the array of the role IDs you wish to assign to the restrictions set:
  * role_id - ID of the role you wish to assign to the restrictions set
* resource_ids - the array of resource IDs you wish to limit under the restrictions set:
resource_id - ID of the resource you wish to limit under the restrictions set
68 Roles

This class manages roles assigned to users. A role itself maintains a set of permissions that gives an access to cloud resources and control panel functionality. You can easily regulate roles (and users in turn) using view/edit/delete options.

68.1 Get List of Roles

To get the list of all the roles available in the system, use the following request:

GET /roles.xml
GET /roles.json

XML Request Example


JSON Request Example


XML Output Example

<roles>
  <role>
    <label>Administrator</label>
    <created_at>2010-05-26T13:34:58Z</created_at>
    <updated_at>2010-07-18T21:16:14Z</updated_at>
    <id>1</id>
    <identifier>admin</identifier>
    <permissions>
      <permission>
        <label>Any action on virtual servers</label>
        <created_at>2010-05-26T13:34:58Z</created_at>
        <updated_at>2010-05-26T13:34:58Z</updated_at>
        <id>1</id>
        <identifier>virtual_machines</identifier>
      </permission>
      ...
    </permissions>
  </role>
</roles>

Where:

roles – an array of all roles with their details and assigned permissions
label – role title
created_at – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at – the date when the role was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
id – role ID
identifier – role identifier
permissions – an array with all the permissions assigned to this role, where:

- label – permission title (permission on an action)
- created_at – time in [YYYY][MM][DD][T][hh][mm][ss]Z format
- updated_at – time in [YYYY][MM][DD][T][hh][mm][ss]Z format
- id – permission ID
- identifier – permission identifier

68.2 Get Role Details

To get the details of a particular user role, use the following request:

GET /roles/:id.xml
GET /roles/:id.json

XML Request Example


JSON Request Example


XML Output Example

```
<role>
  <label>TT</label>
  <created_at type="datetime">2011-02-11T11:20:00Z</created_at>
  <updated_at type="datetime">2011-02-11T13:56:44Z</updated_at>
  <id type="integer">3</id>
  <identifier>gkue74amkiznb7</identifier>
  <permissions type="array">
    <permission>
      <label>Any action Sysadmin Tools</label>
      <created_at type="datetime">2011-02-11T10:35:16Z</created_at>
      <updated_at type="datetime">2011-02-11T10:35:16Z</updated_at>
      <id type="integer">4</id>
      <identifier>sysadmin_tools.read</identifier>
    </permission>
  </permissions>
</role>
```

For details, refer Get List of Roles section.

The role for a particular user is output on /users/:id request.
68.3 Add Role

To add a new role, use the following request:

**POST /roles.xml**
**POST /roles.json**

**XML Request Example**

```bash
curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
  '<role><label>New_role_xml</label><permission_ids type="array"><permission_id>12</permission_id>
  <permission_id>14</permission_id><permission_id>16</permission_id><permission_id>11</permission_id><per
mission_id>10</permission_id><permission_id>35</permission_id></permission_ids></role>'
  --url http://onapp.test/roles.xml
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d
  '{"role":{"label":"New_role_json","permission_ids":[12,14,16,11,10,35]}}'
  --url http://onapp.test/roles.json
```

Where:

- `label` - the new role label
- `permission_ids` - the array of IDs of the permission you would like to assign to this role

68.4 Edit Role

To edit a role, use the following request:

**PUT /roles/:id.xml**
**PUT /roles/:id.json**

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' -d
  '<role><label>changed</label><permission_ids type="array"><permissions_id>12</permissions_id>
  <permissions_id>14</permissions_id><permissions_id>6</permissions_id><permissions_id>1</permissions_id></permission_ids></role>'
  --url http://onapp.test/roles/2.xml
```

**JSON Request Example**

```bash
null
```
curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"role":{"label":"jsonchanged","permission_ids":[1,2,3,4,5,6,7,8,9]}}' --url http://onapp.test/roles/2.json

Where:

*label* – role title

*permission_ids* – ID of permissions, which you want to assign to this role

### 68.5 Delete Role

To delete a user role, use the following request:

DELETE /roles/:id.xml
DELETE /roles/:id.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/roles/2.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/roles/2.json
```

This returns an HTTP 204 response if the role is deleted, or HTTP 404 if the user with the specified ID isn't found.

### 68.6 Edit User Role Assignment

To change a role, assigned to the user, add a new role (or set of roles), use the following request:

PUT /users/:id.xml
PUT /users/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:
role_ids – ID of role(s) you want to assign to the user
This returns an HTTP 204 response if roles are changed, or HTTP 404 if the specified role ID isn’t found.

### 68.7 Get List of All Permissions

To get the list of all available permissions, use the following request:

GET /permissions.xml
GET /permissions.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<permissions>
  <permission>
    <label>Any action on virtual servers</label>
    <created_at>2010-05-26T13:34:58Z</created_at>
    <updated_at>2010-05-26T13:34:58Z</updated_at>
    <id>1</id>
    <identifier>virtual_machines</identifier>
  </permission>
  ...
  <permission></permission>
  ...
</permissions>
```

**Where:**

- `label` – permission title (permission on an action)
- `created_at` – time in [YYYY][MM][DD][hh][mm][ss]Z format
- `updated_at` – time in [YYYY][MM][DD][hh][mm][ss]Z format
- `id` – permission ID
- `identifier` – permission identifier

### 68.8 Clone Role

To clone a role, use the following request:

POST /roles/:id/clone.xml
POST /roles/:id/clone.json
XML Request Example

```
```

JSON Request Example

```
```

Where:

| id | the role ID |

XML Output Example

```
<role>
  <created_at type="datetime">2014-11-08T14:24:33+02:00</created_at>
  <id type="integer">21</id>
  <identifier>b99q74ua00ufw7</identifier>
  <label>Example role - copy(2014-11-08 12:24:33)</label>
  <updated_at type="datetime">2014-11-08T14:24:33+02:00</updated_at>
  <permissions type="array">
    <permission>
      <created_at type="datetime">2014-11-05T13:12:10+02:00</created_at>
      <id type="integer">533</id>
      <identifier>activity_logs</identifier>
      <label>Any action on activity logs</label>
      <updated_at type="datetime">2014-11-05T13:12:10+02:00</updated_at>
    </permission>
  </permissions>
</role>
```

Where:

- **created_at** - the date when the record in the database was created
- **id** - the ID of the role
- **identifier** - the identifier in the database of the role
- **label** - the label of the role, which is by default the "original" role label with the date when the role has been cloned.
- **updated_at** - the date when the record in the database was updated
- **permissions** - the array with the list of permissions for this role
- **id** - the ID of the permission
- **label** - the name of the permission
69 SAML ID Providers

SAML Authentication enables the integration of OnApp as a Service Provider into third-party systems via Single Sign-On possibility, so that users of third-party systems can use their credentials to access OnApp services, without the need to be previously registered in OnApp Cloud.

69.1 Get List of SAML ID Providers

To view the list of SAML identity providers, use the following request:

GET /settings/authentication/saml_id_providers.xml
GET /settings/authentication/saml_id_providers.json

**XML Request Example**

```
curl -i -X GET -u user:password --url
https://onapp.test/settings/authentication/saml_id_providers.xml
```

**JSON Request Example**

```
curl -i -X GET -u user:password --url
https://onapp.test/settings/authentication/saml_id_providers.json
```

**XML Output Example**
<saml_id_providers type="array">
  <saml_id_provider>
    <id type="integer">1</id>
    <name>provider_name</name>
    <issuer>https://service.provider.com/issuer</issuer>
    <icon/>
    <idp_sso_target_url>https://onapp.com/ls/idp_sso_target_url</idp_sso_target_url>
    <idp_cert>-----BEGIN CERTIFICATE-----
    MITC4DCCAqigAwIBAgIQRRWNzx0IzA7VWMYJ3u6vM+iANBgkqhkiG9w0BAQsFADA=MSowKAYDVQQDEyFBREZTIFNpZ25pbmcgLSBhZGZzLm9uYXBwZGV2Lmx2aXYwHhcN
    -----END CERTIFICATE-----
    </idp_cert>
    <enabled type="boolean">true</enabled>
    <assertion_consumer_service_url>https://test.onapp/users/auth/saml/callback?provider_id=1</assertion_consumer_service_url>
    <onapp_key>OnApp_key</onapp_key>
    <user_name_key>un_key</user_name_key>
    <roles_key/>
    <user_group_key/>
    <time_zone_key/>
    <created_at type="dateTime">2017-09-19T15:59:56+03:00</created_at>
    <updated_at type="dateTime">2017-09-20T12:39:17+03:00</updated_at>
    <nameid_format>emailAddress</nameid_format>
    <user_email_key>OnApp_UserEmail</user_email_key>
    <encrypted_assertion type="boolean">false</encrypted_assertion>
    <private_key/>
    <user_billing_plan_key>bp_key</user_billing_plan_key>
    <first_name_key>first_name_key</first_name_key>
    <last_name_key>last_name_key</last_name_key>
    <locale_key/>
    <system_theme_key/>
    <display_infoboxes_key/>
    <disable_auto_suspend_key/>
    <suspend_after_key/>
    <suspend_at_key/>
  </saml_id_provider>
  <saml_id_provider>...
  </saml_id_provider>
  </saml_id_providers>

Where:

id – the ID of the identity provider
name – the name of the identity provider
issuer – the name of the service provider
icon – the icon that is displayed on the login page
idp_sso_target_url – the URL to which the login authentication request is sent
idp_cert – the certificate of the identity provider in the PEM format
enabled – the status of the identity provider that can be enabled if true or disabled if false
assertion_consumer_service_url – the URL of the assertion consumer service
onapp_key – the key of the attribute that enables the import and synchronization of user attributes during every login to OnApp through the IdP instance
roles_key – the key of the role attribute that creates or syncs the user’s role in OnApp
user_group_key – the key of the attribute that assigns the user to a particular user group

time_zone_key – the key of the attribute that assigns the user to a particular time zone

created_at – the date when the identity provider was created

updated_at – the date when the identity provider was updated

nameid_format – the name identifier format according to the Oasis SAML specification

user_email_key – the key of the attribute that provides an email of the user

encrypted_assertion – the status of the encrypted assertion that can be enabled if true or disabled if false

private_key – the private key for the enabled encrypted assertion

user_billing_plan_key – the key of the attribute that assigns the user to a particular bucket under which this user is billed

first_name_key – the key of the attribute that imports the first name of the user

last_name_key – the key of the attribute that imports the last name of the user

locale_key – the key of the attribute that contains the language in which OnApp Cloud UI is available to the user

system_theme_key – the key of the attribute that indicates a default system theme in which OnApp Cloud UI is available to the user

display_infoboxes_key – the key of the attribute that enables or disables the display of infoboxes to the user

disable_auto_suspend_key – the key of the attribute that enables or disables the auto-suspending of the user

suspend_after_key – the key of the attribute that indicates the period of time in hours after which the user will be suspended

suspend_at_key – the key of the attribute that indicates the date and time when the user will be suspended

idp_slo_target_url – the URL to which the logout request is sent

Page History

v.6.0

• removed the idp_cert_fingerprint parameter

69.2 Get SAML ID Provider Details

To view the details of a SAML identity provider, use the following request:

GET /settings/authentication/saml_id_providers/:id.xml

GET /settings/authentication/saml_id_providers/:id.json

XML Request Example

curl -i -X GET -u user:password --url
https://onapp.test/settings/authentication/saml_id_providers/2.xml

JSON Request Example
curl -i -X GET -u user:password --url https://onapp.test/settings/authentication/saml_id_providers/2.json

XML Output Example

```
<onapp_key>OnApp_key</onapp_key>
```

Where:
- `id` – the ID of the identity provider
- `name` – the name of the identity provider
- `issuer` – the name of the service provider
- `icon` – the icon that is displayed on the login page
- `idp_sso_target_url` – the URL to which the login authentication request is sent
- `idp_cert` – the certificate of the identity provider in the PEM format
- `enabled` – the status of the identity provider that can be enabled if `true` or disabled if `false`
- `assertion_consumer_service_url` – the URL of the assertion consumer service
onapp_key – the key of the attribute that enables the import and synchronization of user attributes during every login to OnApp through the IdP instance

roles_key – the key of the role attribute that creates or syncs the user’s role in OnApp

user_group_key – the key of the attribute that assigns the user to a particular user group

time_zone_key – the key of the attribute that assigns the user to a particular time zone

created_at – the date when the identity provider was created

updated_at – the date when the identity provider was updated

nameid_format – the name identifier format according to the Oasis SAML specification

user_email_key – the key of the attribute that provides an email of the user

encrypted_assertion – the status of the encrypted assertion that can be enabled if true or disabled if false

private_key – the private key for the enabled encrypted assertion

user_billing_plan_key – the key of the attribute that assigns the user to a particular bucket under which this user is billed

first_name_key – the key of the attribute that imports the first name of the user

last_name_key – the key of the attribute that imports the last name of the user

locale_key – the key of the attribute that contains the language in which OnApp Cloud UI is available to the user

system_theme_key – the key of the attribute that indicates a default system theme in which OnApp Cloud UI is available to the user

display_infoboxes_key – the key of the attribute that enables or disables the display of infoboxes to the user

disable_auto_suspend_key – the key of the attribute that enables or disables the auto-suspending of the user

suspend_after_key – the key of the attribute that indicates the period of time in hours after which the user will be suspended

suspend_at_key – the key of the attribute that indicates the date and time when the user will be suspended

idp_slo_target_url – the URL to which the logout request is sent

Page History

v.6.0

- removed the idp_cert_fingerprint parameter
70 SDN Management

Software Defined Networking is a technology that simplifies network management via intelligence centralization in just one network component. The feature provides the ability to manage networks using VXLAN technology across OnApp cloud compute resources.

70.1 SDN Manager

SDN manager is an OnApp control panel entity used to manage the SDN infrastructure using API calls for the ODL controller. SDN manager connection options are used to connect SDN Nodes to the ODL controller.

70.1.1 Get SDN Manager Details

To get SDN manager details, use the following request:

GET /settings/sdn/managers/:id.xml
GET /settings/sdn/managers/:id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<networking_sdn_manager>
  <id type="integer">1</id>
  <label>Manager185-Karaf_qah_test</label>
  <host>10.0.25.100</host>
  <port type="integer">8080</port>
  <login>admin</login>
  <password>admin</password>
  <created_at type="dateTime">2018-03-14T12:28:07+00:00</created_at>
  <updated_at type="dateTime">2018-03-26T10:46:35+00:00</updated_at>
</networking_sdn_manager>
```

**Where:**

- *id* - the ID of the SDN manager
- *label* - the name of the SDN manager
- *host* - the hostname or IP address of the ODL controller
- *port* - the port to connect to ODL controller (e.g. 9090, 8080)
- *login* - user login name to login into ODL controller
**password** - user password

**created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss] format

**updated_at** - the date when the SDN manager was updated in the
[YYYY][MM][DD][T][hh][mm][ss] format

### 70.1.2 Add SDN Manager

To create SDN manager, use the following request:

POST /settings/sdn/managers.xml
POST /settings/sdn/managers.json

**XML Request Example**

curl -i -X POST -u user:userpass --url
-H 'Content-type: application/xml' -d
"<networking_sdn_manager><label>Manager</label><host>10.0.51.133</host><login>admin</login><password>admin</password><port>8080</port></networking_sdn_manager>"

**JSON Request Example**

curl -i -X POST -u user:userpass --url
-H 'Content-type: application/json' -d 
{"networking_sdn_manager":
"label": "Manager", "host": "10.0.51.133", "login": "admin", "password": "admin", "port": "8080"}

Where:

- **label** - the name of the SDN manager
- **host** - the hostname or IP address of the ODL controller
- **port** - the port to connect to ODL controller (e.g. 9090, 8080)
- **login** - user login name to login into ODL controller
- **password** - user password

### 70.1.3 Edit SDN Manager

To edit SDN manager, use the following request:

PUT /settings/sdn/managers/:id.xml
PUT /settings/sdn/managers/:id.json

**XML Request Example**

curl -i -X PUT -u user:userpass --url
http://onapp.test/settings/sdn/managers/74.xml -H 'Accept: application/xml'
-H 'Content-type: application/xml' -d 
"<networking_sdn_manager><label>Manager</label><host>10.0.51.145</host><login>admin</login><password>admin</password><port>8080</port></networking_sdn_manager>"

**JSON Request Example**

curl -i -X PUT -u user:userpass --url
http://onapp.test/settings/sdn/managers/74.json -H 'Accept: application/json'
-H 'Content-type: application/json' -d 
{"networking_sdn_manager":
"label": "Manager", "host": "10.0.51.145", "login": "admin", "password": "admin", "port": "8080"}
### JSON Request Example

```
```

**Where:**

- **label** - the name of the SDN manager
- **host** - the hostname or IP address of the ODL controller
- **login** - user login name to login into ODL controller
- **password** - user password
- **port** - the port to connect to ODL controller (e.g. 9090, 8080)

### 70.1.4 Delete SDN Manager

To delete SDN manager, use the following request:

DELETE /settings/sdn/managers/:id.xml

DELETE /settings/sdn/managers/:id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

### 70.1.5 Add Connection Option to SDN Manager

To add connection option to SDN manager, use the following request:

POST /sdn/managers/:id/connection_options.xml

POST /sdn/managers/:id/connection_options.json

**XML Request Example**

```
```

**JSON Request Example**

```
```
OnApp Cloud 6.3 API Guide

70.1.6 Delete Connection Option from SDN Manager

To delete connection option from SDN manager, use the following request:

DELETE /sdn/managers/:id/connection_options/:id.xml
DELETE /sdn/managers/:id/connection_options/:id.json

XML Request Example

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/76/connection_options/22.xml

-H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/76/connection_options/22.json

-H 'Accept: application/json'
-H 'Content-type: application/json'
```

70.2 SDN Nodes

A node is an OpenVSwitch instance installed by OnApp installation script on compute resources. You can assign and unassign nodes that have OpenVSwitch installed to be managed by SDN manager. It will allow you to select the compute resource from compute resource zones.

70.2.1 Get List of SDN Manager Nodes

To get a list of SDN manager nodes, use the following request:

GET /settings/sdn/managers/:id/nodes.xml
GET /settings/sdn/managers/:id/nodes.json

XML Request Example

```bash
curl -i -X GET -u user:userpass --url
http://onapp.test/settings/sdn/managers/1/nodes.xml

-H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass --url
http://onapp.test/settings/sdn/managers/1/nodes.json

-H 'Accept: application/json'
-H 'Content-type: application/json'
```
XML Output Example

```
<networking_sdn_nodes type="array">
  <networking_sdn_node>
    <id type="integer">7</id>
    <compute_resource_id type="integer">7</compute_resource_id>
    <connection_option_id type="integer">1</connection_option_id>
    <system_id>7f2011fc-5cad-4d77-a239-dce8b646ace6</system_id>
    <status>connected</status>
    <created_at type="dateTime">2018-03-15T15:08:42+00:00</created_at>
    <updated_at type="dateTime">2018-03-15T15:08:48+00:00</updated_at>
  </networking_sdn_node>
  ...
</networking_sdn_nodes>
```

Where:
- **id**: compute zone ID
- **compute_resource_id**: the ID of the compute resource
- **connection_option_id**: the ID of the connection option
- **system_id**: the ID of the system
- **status**: status of connection
- **created_at**: time when the node was created, in [YYYY][MM][DD][hh][mm][ss] format
- **updated-at**: time when the node was updated, in [YYYY][MM][DD][hh][mm][ss] format

### 70.2.2 Add Nodes to SDN Manager

To add nodes to SDN manager, use the following request:

**XML Request Example**

```
```

**JSON Request Example**

```
```
curl -i -X POST -u user:userpass --url

Where:

- **manager_id** - the ID of the SDN manager
- **connection_option_id** - the ID of the connection option
- **compute_resource_id** - the ID of the compute resource

### 70.2.3 Reattach SDN Node

To reattach the node, use the following request:

DELETE /settings/sdn/managers/:manager_id/nodes/:node_id/reattach.xml
DELETE /settings/sdn/managers/:manager_id/nodes/:node_id/reattach.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/43/nodes/159/reattach
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/43/nodes/159/reattach
-H 'Accept: application/json' -H 'Content-type: application/json'
```

### 70.2.4 Delete Nodes from SDN Manager

To remove nodes from SDN manager, use the following request:

DELETE /settings/sdn/managers/:manager_id/nodes/:node_id.xml
DELETE /settings/sdn/managers/:manager_id/nodes/:node_id.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/77/nodes/43.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/77/nodes/43.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```
70.3 SDN Networks

An SDN network consists of Open vSwitch (OVS) bridges interconnected using VXLAN Tunnel End Points. These bridges are created on compute resources selected during SDN network creation process.

70.3.1 Get List of SDN Networks

To get a list of SDN networks, use the following request:

GET /settings/sdn/managers/:id/networks.xml
GET /settings/sdn/managers/:id/networks.json

XML Request Example


JSON Request Example


XML Output Example

<networking_sdn_networks type="array">
  <networking_sdn_network>
    <id type="integer">1</id>
    <label>testSdnNet</label>
    <identifier>niyuutaedilvhs</identifier>
    <external_network_id type="integer">1684</external_network_id>
    <vni type="integer">101</vni>
    <manager_id type="integer">1</manager_id>
    <created_at type="dateTime">2018-03-15T15:11:55+00:00</created_at>
    <updated_at type="dateTime">2018-03-15T15:11:55+00:00</updated_at>
  </networking_sdn_network>
  ...
</networking_sdn_networks>

Where:

- **id** - SDN network ID
- **label** - the name of the SDN network
- **identifier** - SDN network Identifier
- **external_network_id** - the ID of the external network
- **vni** - VXLAN Network Identifier (or VXLAN Segment ID)
- **manager_id** - the ID of the SDN manager
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss] format
- **updated_at** - the date when the event was updated in the [YYYY][MM][DD][hh][mm][ss] format
**status** - connection status. Can be initial, connected or failed

### 70.3.2 Get SDN Network Details

To get SDN network details, use the following request:

GET /settings/sdn/managers/:manager_id/networks/:id.xml

GET /settings/sdn/managers/:manager_id/networks/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**

```
<networking_sdn_network>
  <id type="integer">1</id>
  <label>testSdnNet</label>
  <identifier>niyuutaedilvhs</identifier>
  <external_network_id type="integer">1684</external_network_id>
  <vni type="integer">101</vni>
  <manager_id type="integer">1</manager_id>
  <created_at type="dateTime">2018-03-15T15:11:55+00:00</created_at>
  <updated_at type="dateTime">2018-03-15T15:11:55+00:00</updated_at>
  <status>initial</status>
</networking_sdn_network>
```

**Where:**

- **id** - SDN network ID
- **label** - the name of the SDN network
- **identifier** - SDN network Identifier
- **external_network_id** - the ID of the external network
- **vni** - VXLAN Network Identifier
- **manager_id** - the ID of the SDN manager
- **created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **updated_at** - the date when the event was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **status** - connection status. Can be initial, connected or failed
70.3.3 Add SDN Network

To add new SDN network, use the following request:

POST /settings/sdn/managers/:id/networks.xml
POST /settings/sdn/managers/:id/networks.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:

- **label** - the name of the SDN network
- **manager_id** - the ID of the SDN manager
- **vni** - VXLAN Network Identifier
- **id** - the ID of the node
- **default_connection_ip** - the IP address of the node connection
- **network_id** - the ID of the external network

70.3.4 Connect SDN Network to SDN Node

To connect SDN network to SDN nodes, use the following request:

POST /settings/sdn/managers/:manager_id/networks/:id/bridges.xml
POST /settings/sdn/managers/:manager_id/networks/:id/bridges.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```
### JSON Request Example

```bash
```

**Where:**
- `manager_id` - the ID of the manager
- `network_id` - the ID of the SDN network
- `connecting_node_id` - the ID of the connecting node
- `node_id` - the ID of the node to which SDN network is connected
- `local_ip` - IP address of the node
- `remote_ip` - IP address of the connecting node

#### 70.3.5 Assign SDN Network to User

To assign an SDN network to a particular user, use the following request:

POST /settings/networks/:network_id/assign_to_user.xml

POST /settings/networks/:network_id/assign_to_user.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**Where:**
- `user_id` - the ID of the user to which you assign an SDN network

#### 70.3.6 Unassign SDN Network from User

To unassign an SDN network from a particular user, use the following request:

DELETE /settings/networks/:network_id/unassign_from_user.xml

DELETE /settings/networks/:network_id/unassign_from_user.json

**XML Request Example**

```bash
```

**JSON Request Example**

curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/settings/networks/7/unassign_from_user.json -d '{"user_id":5}"

Where:

`user_id` - the ID of the user to which you assign an SDN network

### 70.3.7 Delete SDN Network

To delete SDN network, use the following request:

DELETE /settings/sdn/managers/:manager_id/networks/:id.xml
DELETE /settings/sdn/managers/:manager_id/networks/:id.json

**XML Request Example**


**JSON Request Example**


### 70.3.8 Recreate Bridges

To recreate SDN bridge, use the following request:

PUT /settings/sdn/managers/:manager_id/networks/:network_id/bridges/:bridge_id/recreate.xml
PUT /settings/sdn/managers/:manager_id/networks/:network_id/bridges/:bridge_id/recreate.json

**XML Request Example**

### JSON Request Example

```
curl -i -X PUT -u 'user:userpass' --url
http://onapp.test/settings/sdn/managers/1/networks/6/bridges/25/recreate.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

### 70.3.9 Delete Bridge

To delete SDN bridge, use the following request:

```
DELETE
/settings/sdn/managers/:manager_id/networks/:network_id/bridges/:bridge_id.xml
```
```
DELETE
/settings/sdn/managers/:manager_id/networks/:network_id/bridges/:bridge_id.json
```

#### XML Request Example

```
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/43/networks/16/bridges/60.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

#### JSON Request Example

```
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/sdn/managers/43/networks/16/bridges/60.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

### 70.3.10 Cleanup Zombie Tunnels

To cleanup zombie tunnels, use the following request:

```
POST
/settings/sdn/managers/:manager_id/networks/:network_id/cleanup_zombie_tunnels.xml
```
```
POST
/settings/sdn/managers/:manager_id/networks/:network_id/cleanup_zombie_tunnels.json
```

#### XML Request Example

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/settings/sdn/managers/100/networks/79/cleanup_zombie_tunnels
```

#### JSON Request Example

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/settings/sdn/managers/100/networks/79/cleanup_zombie_tunnels
```
71 Smart Servers

Smart servers are dedicated entities based on KVM CloudBoot compute resources with passthrough enabled.

NOTE: VLANs are not configured automatically on smart servers. You need to configure them manually in accordance with your OS and hardware settings.

Smart servers are billed the same way as Virtual Servers. You can set limits and prices for CPU/CPU share/memory.

To charge for smart server resources:
1. Create a smart server compute zone
2. Attach smart server compute resources to this zone
3. Add this compute zone (smart server type) to a bucket and set the CPU/CPU share/memory limits
4. Assign user to this bucket
5. Create a smart server under this user’s account and allocate the required smart server on a compute zone that you’ve just added to the bucket.

Currently, it is not possible to set limits and prices for smart server network resources.

71.1 Get List of Smart Servers

To get the list of Smart Servers, use the following request:

GET /smart_servers.xml
GET /smart_servers.json

XML Request Example


JSON Request Example


XML Output Example
<smart_server>
  <add_to_marketplace nil="true"/>
  <admin_note nil="true"/>
  <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
  <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
  <allowed_swap type="boolean">true</allowed_swap>
  <booted type="boolean">true</booted>
  <built type="boolean">true</built>
  <cpu_shares type="integer">1</cpu_shares>
  <cpus type="integer">1</cpus>
  <created_at type="datetime">2013-07-25T12:11:09+03:00</created_at>
  <customer_network_id nil="true"/>
  <deleted_at nil="true"/>
  <edge_server_type nil="true"/>
  <enable_autoscale type="boolean">false</enable_autoscale>
  <enable_monitis type="boolean">false</enable_monitis>
  <firewall_notrack type="boolean">false</firewall_notrack>
  <hostname>test</hostname>
  <hypervisor_id type="integer">67</hypervisor_id>
  <id type="integer">103</id>
  <identifier>on8n97e0g0gfci</identifier>
  <initial_root_password>qweasdzxc</initial_root_password>
  <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
  <label>TT</label>
  <locked type="boolean">false</locked>
  <memory type="integer">384</memory>
  <min_disk_size type="integer">5</min_disk_size>
  <note nil="true"/>
  <operating_system>linux</operating_system>
  <operating_system_distro>rhel</operating_system_distro>
  <preference_distro type="array">
    <fixnum type="integer">1</fixnum>
  </preference_distro>
  <recovery_mode nil="true"/>
  <remote_access_password>M8eZXy9zJQhP</remote_access_password>
  <service_password nil="true"/>
  <state>new</state>
  <storage_server_type nil="true"/>
  <strict_virtual_machine_id nil="true"/>
  <suspended type="boolean">false</suspended>
  <template_id type="integer">2</template_id>
  <template_label>CentOS 6.4 x64 1.3</template_label>
  <time_zone>Atlantic Time (Canada)</time_zone>
  <updated_at type="datetime">2013-07-25T12:14:15+03:00</updated_at>
  <user_id type="integer">6</user_id>
  <vip nil="true"/>
  <xen_id type="integer">2</xen_id>
</smart_server>

<local_remote_access_ip_address>83.170.81.132</local_remote_access_ip_address>
<local_remote_access_port type="integer">5900</local_remote_access_port>
<locked type="boolean">false</locked>
<min_disk_size type="integer">5</min_disk_size>
<note nil="true"/>
<operating_system>linux</operating_system>
<operating_system_distro>rhel</operating_system_distro>
<preference_distro type="array">
  <fixnum type="integer">1</fixnum>
</preference_distro>
<recovery_mode nil="true"/>
<remote_access_password>M8eZXy9zJQhP</remote_access_password>
<service_password nil="true"/>
</state>

<state>new</state>

<storage_server_type nil="true"/>
<strict_virtual_machine_id nil="true"/>
<suspended type="boolean">false</suspended>
<template_id type="integer">2</template_id>
<template_label>CentOS 6.4 x64 1.3</template_label>
<time_zone>Atlantic Time (Canada)</time_zone>
<updated_at type="datetime">2013-07-25T12:14:15+03:00</updated_at>
<user_id type="integer">6</user_id>
<vip nil="true"/>
<xen_id type="integer">2</xen_id>

<ip_addresses type="array">
  <ip_address>83.170.81.146</ip_address>
  <broadcast>83.170.81.159</broadcast>
  <created_at type="datetime">2013-06-11T10:19:46+03:00</created_at>
  <customer_network_id nil="true"/>
  <disallowed_primary type="boolean">false</disallowed_primary>
  <gateway>83.170.81.145</gateway>
  <hypervisor_id nil="true"/>
  <id type="integer">3</id>
  <ip_address_pool_id nil="true"/>
  <network_address>83.170.81.144</network_address>
  <network_id type="integer">1</network_id>
</ip_addresses>
Where:

add_to_marketplace — empty for Smart Servers; used for edge servers only

admin_note — an optional note of the administrator

allow_resize_without_reboot — true if resize without reboot is possible; otherwise false

allowed_hot_migrate — true if the template, on which the Smart Server is based, supports hot migration; otherwise false

allowed_swap — true if swap disk is allowed (depends on the template the Smart Server is based on); otherwise false

booted — true if the Smart Server is running, otherwise false

built — true if the Smart Server is built, otherwise false

cpus — the number of allocated CPU cores

cpu_shares — CPU priority in percents

created_at — the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

enable_autoscale — true if autoscaling is allowed for this Smart Server

hostname — the name of your host

hypervisor_id — the ID of the compute resource used by this Smart Server

id — the Smart Server ID

identifier — the Smart Server identifier

initial_root_password — the VS root password

initial_root_password_encrypted - true, if the root password is encrypted, otherwise false.

ip_addresses — an array of ip addresses with their details assigned to this Smart Server

label — the Smart Server label

local_remote_access_port — the port ID used for console access

locked — true if the Smart Server is locked; otherwise false

max_memory — maximum amount of RAM which can be allocated to the Smart Server by the compute resource

memory — the RAM size allocated to this Smart Server
**min_disk_size** — the minimum disk size required to build a Smart Server from a specified template

**primary_disk_min_iops** - minimum number of IO operations per second for primary disk (this is a SolidFire related parameter)

**swap_disk_min_iops** - minimum number of IO operations per second for swap disk (this is a SolidFire related parameter)

**note** — an optional reminder for this Smart Server made by a user account

**network_address** – the address of the network

**operating_system** — operating system used by the Smart Server

**operating_system_distro** — the distribution of the OS from which this Smart Server is built

**recovery_mode** — true if recovery mode allowed. Otherwise false

**remote_access_password** — the password for the remote access

**state** – parameter reserved for future use

**strict_virtual_machine_id** — the ID of a virtual server that will never reside on the same compute resource with this Smart Server

**suspended** — true if Smart Server is suspended, otherwise false

**template_id** — the ID of the template the Smart Server is based on

**template_label** — the name of the template from which this Smart Server is built

**time_zone** - the time zone set for the smart server. This parameter is applicable only to Windows smart servers

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows smart server manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a smart server after starting it if time synchronization is not completed for some reason.

**total_disk_size** — the total disk size in GB of all disks assigned to Smart Server

**updated_at** — the date when the Smart Server was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**user_id** — the ID of a user assigned to this Smart Server

**vip** — true if the Smart Server has VIP status (gives migration priority)

**xen_id** — the Smart Server ID set by the virtualization engine

**cpu_priority** - this is a new parameter reserved for future use; currently it has the same value as **cpu_shares** parameter

### 71.2 Get Smart Server Details

GET /smart_servers/:id.xml
GET /smart_servers/:id.json

Shows the same attributes of the Smart Servers described in [Get List of Smart Servers](#) request.
71.3 Add Smart Server

To add a new smart server, use the following request:

POST /smart_servers.xml
POST /smart_servers.json

**XML Request Example**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-Type: application/xml' -d
  '<smart_server><template_id>1</template_id><licensing_key></licensing_key>
  <label>zaza(XML)</label><hostname>zaza</hostname><domain>localdomain</domain>
  <hypervisor_id>355</hypervisor_id><initial_root_password>qwaszx</initial_root_password><initial_root_password_confirmation>qwaszx</initial_root_password_confirmation><memory>512</memory><cpus>1</cpus><cpu_shares>1</cpu_shares><data_store_group_primary_id>14</data_store_group_primary_id><primary_disk_size>5</primary_disk_size><prefer_local_reads>1</prefer_local_reads><data_store_group_swap_id>14</data_store_group_swap_id><swap_disk_size>1</swap_disk_size><primary_network_group_id>3</primary_network_group_id><rate_limit>1</rate_limit><required_ip_address_assignment>1</required_ip_address_assignment><required_automatic_backup>0</required_automatic_backup><required_virtual_machine_build>1</required_virtual_machine_build><required_virtual_machine_startup>1</required_virtual_machine_startup><time_zone>Atlantic Time (Canada)</time_zone><enable_autoscale>0</enable_autoscale><recipe_ids type='array'><recipe_id>11</recipe_id></recipe_ids><custom_recipe_variables><custom_recipe_variable><name="varname" value="var_value" enabled="1"></custom_recipe_variable></custom_recipe_variables></smart_server>'
```

**JSON Request Example**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-Type: application/json' -d
  '{"smart_server":{"template_id":"1","licensing_key":""","label":"zaza(JSON)","hostname":"zaza","domain":"localdomain","hypervisor_id":"355","initial_root_password":"qwaszx","initial_root_password_confirmation":"qwaszx","memory":"512","cpus":"1","cpu_shares":"1","data_store_group_primary_id":"14","primary_disk_size":"5","prefer_local_reads":"1","data_store_group_swap_id":"14","swap_disk_size":"1","primary_network_group_id":"3","rate_limit":"1","required_ip_address_assignment":"1","time_zone":"Atlantic Time (Canada)","required_automatic_backup":"0","required_virtual_machine_build":"1","enable_autoscale":"0","recipe_ids":"array","recipe_id":11","custom_recipe_variables":"{"custom_recipe_variable":{"name":"varname","value":"var_value","enabled":"1"}}"}}'
```

Where:

- `template_id` * - the ID of a template from which a smart server should be built
- `licensing_key` - the key of a license. This parameter is for Windows servers only
- `label` * - user-friendly smart server description
- `hostname` * - specify the smart server hostname
- `domain` - specify the domain for this VS. The default value is `localdomain`. This parameter is not applicable for Windows virtual servers.
**hypervisor_id** - the ID of a smart compute resource where the smart server will be built. If no smart compute resource ID is specified, the server will be built on the compute resource with the least available RAM (but sufficient RAM for the server).

**initial_root_password** - the root password for a smart server. If none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _ ], and the following special characters: ~ ! @ # $ ^ & * ( ) = _ + \ | \ { } [ ] : ; ' . ? / . You can use both lower- and uppercase letters.

**initial_root_password_confirmation** - the root password confirmation

**primary_network_group_id** - the ID of the primary network group. Optional parameter

**required_ip_address_assignment** - set "1" if you want IP address to be assigned automatically after creation. Otherwise set "0"

**memory** * - amount of RAM assigned to the smart server

**cpus** * - number of CPUs assigned to the smart server. For KVM compute resources, this parameter sets CPU sockets by default, unless CPU topology is enabled.

**cpu_shares** - optional parameter. For KVM compute resource the CPU priority value is always 100. For XEN, set a custom value. The default value for XEN is 1

**cpu_units** - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket.

**cpu_sockets** - the amount of CPU sockets. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted.

**cpu_threads** - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted.

**data_store_group_primary_id** - set the ID of the data store zone to which this primary disk is allocated

**data_store_group_swap_id** - set the ID of the data store zone to which this swap disk is allocated

**primary_disk_size** * - set the disk space for this smart server

**swap_disk_size** * - set swap space. There is no swap disk for Windows-based smart servers

**primary_network_group_id** - the ID of the primary network group. Optional parameter

**rate_limit** - set max port speed. Optional parameter: if none set, the system sets port speed to unlimited

**time_zone** - the time zone set for the smart server. This parameter is applicable only to Windows smart servers

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows smart server manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a smart server after starting it if time synchronization is not completed for some reason.

**required_automatic_backup** - set 1 if you need automatic backups.

**required_virtual_machine_build** * - set 1 to build smart server automatically

**required_virtual_machine_startup** - set 1 to start up the smart server automatically, otherwise set 0 (default state is "1")
**required_ip_address_assignment** - set "1" if you want container server to be created with already assigned IP address, otherwise set "0"; IP address can be assigned after container server creation.

**selected_ip_address** - an IP address to assign to this container server; if the parameter **required_ip_address_assignment** was set "1" but this parameter **selected_ip_address** is empty - the first available IP address will be assigned to container server automatically.

**enable_autoscale** - set 1 to enable autoscale, otherwise set 0

**recipe_ids** - an array of recipe ID you want to run on the smart server provisioning

**custom_variables** - an array of custom variables with the following details:

- **enabled** - true, if the variable is enabled, otherwise false
- **id** - variable ID
- **name** - variable name
- **value** - variable value script

**Page History**

v.5.4

- added the following parameters:
  - **domain**
  - **selected_ip_address**

### 71.4 View Encrypted Smart Server Password

If the Smart Server was created with password encryption enabled, you can use the following API call to view the password (the request returns the decrypted password).

To view the encrypted smart server password, use the following request:

**XML Request Example:**

```
curl -X GET -u user:userpass http://onapp.test/smart_servers/9/with_decrypted_password.xml?initial_root_password_encryption_key=encryptionkey
```

**JSON Request Example:**

```
curl -X GET -u user:userpass http://onapp.test/smart_servers/9/with_decrypted_password.json?initial_root_password_encryption_key=encryptionkey
```

**Where:**

- **id** – the Smart Server ID

### 71.5 Build Smart Server

To build or re-build a Smart Server, use the following request:
POST /smart_servers/:smart_server_id/build.xml
POST /smart_servers/:smart_server_id/build.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:

- `template_id*` - the ID of a template from which a Smart Server should be built.
- `required_startup` - set to 1 if you wish to start a Smart Server after it is built. Otherwise set to 0.

Instead of Smart Server ID (`:smart_server_id`) you may use Smart Server identifier (`:smart_server_identifier`).

### 71.6 Edit Smart Server

To edit a smart server, use the following request:

PUT /smart_servers/:id.xml
PUT /smart_servers/:id.json

**XML Request Example**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<?xml version="1.0" encoding="UTF-8"?><smart_server><label>Test_API_Edit</label><memory>512</memory><cpu_shares>40</cpu_shares><cpus>4</cpus><allow_migration>1</allow_migration><allow_cold_resize>1</allow_cold_resize><time_zone>Atlantic Time (Canada)</time_zone><primary_disk_min_iops>600</primary_disk_min_iops><swap_disk_min_iops>600</swap_disk_min_iops></smart_server>' --url http://onapp.test/smart_servers/9.xml
```

**JSON Request Example**

```bash
```
853

Where:

- **label** - the Smart Server name
- **memory** - the amount of RAM allocated to this Smart Server in Mb
- **cpus** - the number of CPUs of this Smart Server
- **cpu_shares** - CPU priority percentage
- **allow_migration** - set 1 to migrate a Smart Server to a compute resource with sufficient resources if a compute resource has insufficient space to resize. Otherwise, set 0.
- **allow_cold_resize** – set 1 to switch to cold resize when hot resize failed
- **time_zone** - the time zone set for the smart server. This parameter is applicable only to Windows smart servers.

After you edit the server’s time zone, you need to stop and then start up the smart server.

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows smart server manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a smart server after starting it if time synchronization is not completed for some reason.

- **primary_disk_min_iops** - minimum number of IO operations per second for primary disk (this is a SolidFire related parameter)
- **swap_disk_min_iops** - minimum number of IO operations per second for swap disk (this is a SolidFire related parameter)

If the Smart Server is modified successfully, an HTTP 204 response is returned. If scheduling for changes fails, an HTTP 422 response is returned.

### 71.7 Change Smart Server Owner

To reassign a Smart Server to another user, use the following request:

- POST /smart_servers/:smart_server_id/change_owner.xml
- POST /smart_servers/:smart_server_id/change_owner.json

**XML Request Example**

```xml
<smart_server_change_owner_request>
  <smart_server_id>9</smart_server_id>
  <new_owner_user_id>123456</new_owner_user_id>
  <new_owner_password>password</new_owner_password>
</smart_server_change_owner_request>
```
### 71.8 Migrate Smart Server

To migrate a smart server to another compute resource, use the following request:

**POST** /smart_servers/:smart_server_id/migrate.xml
**POST** /smart_servers/:smart_server_id/migrate.json

**XML Request Example**

```
```

**JSON Request Example**

```
```
Where:

destination* - the ID of a target compute resource where you migrate a smart server

You can only cold migrate smart servers.

71.9 Delete Smart Server

To delete a smart server from the cloud, use the following request:

DELETE /smart_servers/:id.xml
DELETE /smart_servers/:id.json

XML Request Example

```
curl -i -X DELETE -u user:userpass
http://onapp.test/smart_servers/9.xml?convert_last_backup=1&destroy_all_backups=1
```

JSON Request Example

```
curl -i -X DELETE -u user:userpass
http://onapp.test/smart_servers/9.json?convert_last_backup=1&destroy_all_backups=1
```

Where:

id – the ID of a Smart Server you want to delete

convert_last_backup – set 1 to convert the last Smart Server's backup to template, otherwise set 0

destroy_all_backups – set 1 to destroy all existing backups of this Smart Server, otherwise set 0

71.10 Start up Smart Server

When you start up a smart server, it might be implicitly cold migrated if the current compute resource does not have sufficient resources. For more information, refer to Server Provisioning.

To start up a smart server, use the following request:

POST /smart_servers/:smart_server_id/startup.xml
POST /smart_servers/:smart_server_id/startup.json

XML Request Example

```
curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/19/startup.xml
```

JSON Request Example
71.11 Reboot Smart Server

To reboot a smart server, use the following request:

POST /smart_servers/:smart_server_id/reboot.xml
POST /smart_servers/:smart_server_id/reboot.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/10/reboot.xml
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/10/reboot.json
```

An HTTP 201 response is returned on a successful reboot. Unsuccessful reboot responses include HTTP 404 (resource not found – e.g. if the VS isn't online) and HTTP 422 (request cannot be processed – for example, if parameters were incorrect).

71.12 Reboot Smart Server in Recovery

To reboot a smart server in recovery mode with a temporary login ("root") and password ("recovery"), use the following request:

POST /smart_servers/:smart_server_id/reboot.xml
POST /smart_servers/:smart_server_id/reboot.json

**XML Request Example**

```
curl -i -X POST -u user:userpass '<mode>recovery</mode>' --url
http://onapp.test/smart_servers/12/reboot.xml
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass '{"mode":"recovery"}' --url
http://onapp.test/smart_servers/12/reboot.json
```

71.13 Suspend Smart Server

To suspend a smart server, use the following request:

```
curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/19/startup.json
```
OnApp Cloud 6.3 API Guide

POST /smart_servers/:id/suspend.xml
POST /smart_servers/:id/suspend.json

XML Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/12/suspend.xml

JSON Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/12/suspend.json

Where:

smart_server_id* - ID of a smart server you want to suspend

71.14 Unsuspend Smart Server

To activate a smart server again, use the same request as to suspend it:
POST /smart_server/:id/suspend.xml
POST /smart_server/:id/suspend.json

For details refer to Suspense Smart Server section.

71.15 Unlock Smart Server

To unlock a smart server, use the following request:
POST /smart_servers/:smart_server_id/unlock.xml
POST /smart_servers/:smart_server_id/unlock.json

XML Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/12/unlock.xml

JSON Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/smart_servers/12/unlock.json

71.16 Shut down Smart Server

To shut down a smart server, use the following request:
POST /smart_servers/:smart_server_id/shutdown.xml
POST /smart_servers/:smart_server_id/shutdown.json

XML Request Example
71.17 Stop Smart Server

To stop a smart server, use the following request:

```plaintext
POST /smart_servers/:smart_server_id/stop.xml
POST /smart_servers/:smart_server_id/stop.json
```

**XML Request Example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/smart_servers/9/stop.xml
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url http://onapp.test/smart_servers/9/stop.json
```

71.18 Open Smart Server Console

To open a smart server console:

1. Run the following request:

   ```plaintext
   GET /smart_servers/:smart_server_id/console.xml
   GET /smart_servers/:smart_server_id/console.json
   ```

2. Find and copy the value for the `remote_key` parameter in the response output.

3. Open the following URL in the browser:
   ```plaintext
   http://onapp.test/console_remote/[remote_key_parameter_value]
   ```

71.19 Smart Server Autoscaling

SS autoscaling allows you to automatically increase the RAM, CPU and disk size of a smart server. SS resources autoscaling is based on the rules you specify. For example, you can set up a rule that will add 1000 MB of memory to a SS if RAM has been above 90% for the last 10 minutes - but add no more than 5000 MB in total in 24 hours.
71.19.1 Get the List of Autoscaling Rules for Smart Server

To get the list of autoscaling rules for a particular smart server:

GET /smart_servers/:smart_server_id/auto_scaling.xml
GET /smart_servers/:smart_server_id/auto_scaling.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<auto_scaling_configurations type="array">
<auto_scaling_configuration>
<up_to type="integer">9000</up_to>
<for_minutes type="integer">5</for_minutes>
<above type="integer">90</above>
<created_at type="datetime">2011-07-19T18:56:57+07:00</created_at>
<updated_at type="datetime">2011-07-19T18:56:57+07:00</updated_at>
<resource>memory</resource>
<id type="integer">1</id>
<virtual_machine_id type="integer">1063</virtual_machine_id>
<add_units type="integer">600</add_units>
<allow_decrease type="boolean">false</allow_decrease>
</auto_scaling_configuration>
...
</auto_scaling_configuration>
```

Where:
- `up_to` - the amount of resource which cannot be exceeded within 24 hours period
- `for_minutes` - the time threshold before scaling will be triggered
- `above` - the amount of resource usage (%). If this value is reached by the VS during the period specified by the `for_minutes` parameter, the system will add the amount of units set by the `add_units` parameters
- `created_at` - the date when the record in DB was created
- `updated_at` - the date when the record in DB was updated
- `resource` - the resource for which the rule is created (memory/CPU/disk)
- `id` - the ID of the rule
- `virtual_machine_id` - the ID of the SS to which this rule applies
- `add_units` - the amount of resource units which the system should add if the rule is met
- `allow_decrease` - true, if autoscale down is enabled, otherwise false
71.19.2 Create Autoscaling Rule for SS

To create autoscaling rule for a smart server, use the following request:

POST /smart_servers/:smart_server_id/auto_scaling.xml
POST /smart_servers/:smart_server_id/auto_scaling.json

XML Request Example

curl -X POST -u user:userpass
http://onapp.test/smart_servers/12/auto_scaling.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml' -d "<auto_scaling_configuration><up_to>22</up_to><for_minutes>10</for_minutes><above>5</above><resource>cpu</resource><allow_decrease>1</allow_decrease><add_units>22</add_units><enabled>1</enabled><allow_cold_resize>1</allow_cold_resize></auto_scaling_configuration>"

JSON Request Example

curl -X POST -u user:userpass
http://onapp.test/smart_servers/12/auto_scaling.json -H 'Accept: application/json' -H 'Content-Type: application/json' -d ""auto_scaling_configuration":{"above":"5","for_minutes":"10","up_to":"11","resource":"cpu","allow_decrease":"1","add_units":"22","enabled":"1","allow_cold_resize":"1"}""
This will delete all autoscaling rules set for this smart server.

### 71.20 Smart Server Billing Statistics

To view the billing statistics for a particular smart server, use the following request:

GET /smart_servers/:smart_server_id/vm_stats.xml
GET /smart_servers/:smart_server_id/vm_stats.json

To get a shorter statistics output, add an `id` parameter in the URL:

GET /smart_servers/:smart_server_id/vm_stats/:vm_stats_id.xml
GET /smart_servers/:smart_server_id/vm_stats/:vm_stats_id.json

Define a shorter period by setting Start and End time in the API call:


XML Request Example


XML Output Example
<vm_stats>
  <created_at type="datetime">2013-05-02T06:00:27Z</created_at>
  <currency_code>USD</currency_code>
  <id type="integer">15490</id>
  <stat_time type="datetime">2013-05-02T06:00:00Z</stat_time>
  <updated_at type="datetime">2013-05-02T06:00:27Z</updated_at>
  <user_id type="integer">307</user_id>
  <virtual_machine_id type="integer">1214</virtual_machine_id>
  <vm_billing_stat_id type="integer">8089</vm_billing_stat_id>
  <billing_stats><disks type="array">
    <disk>
      <id type="integer">2430</id>
      <costs type="array">
        <cost>
          <value type="integer">100</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_min_iops</resource_name>
        </cost>
        <cost>
          <value type="integer">5</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_size</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_read</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_written</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>reads_completed</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>writes_completed</resource_name>
        </cost>
      </costs>
      <label nil="true"/>
    </disk>
    <disk>
      <id type="integer">2431</id>
      <costs type="array">
        <cost>
          <value type="integer">100</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_min_iops</resource_name>
        </cost>
        <cost>
          <value type="integer">1</value>
          <cost type="float">0.0</cost>
          <resource_name>disk_size</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_read</resource_name>
        </cost>
        <cost>
          <value type="integer">0</value>
          <cost type="float">0.0</cost>
          <resource_name>data_written</resource_name>
        </cost>
      </costs>
    </disk>
  </billing_stats>
</vm_stats>
<cost>
  <value type="integer">0</value>
  <cost type="float">0.0</cost>
  <resource_name>reads_completed</resource_name>
</cost>
</cost>
</disks>
<network_interfaces type="array">
<network_interface>
  <id type="integer">1301</id>
  <costs type="array">
    <cost>
      <value type="integer">1</value>
      <cost type="float">0.0</cost>
      <resource_name>ip_addresses</resource_name>
    </cost>
    <cost>
      <value type="integer">1</value>
      <cost type="float">0.0</cost>
      <resource_name>rate</resource_name>
    </cost>
    <cost>
      <value type="integer">0</value>
      <cost type="float">0.0</cost>
      <resource_name>data_received</resource_name>
    </cost>
    <cost>
      <value type="integer">0</value>
      <cost type="float">0.0</cost>
      <resource_name>data_sent</resource_name>
    </cost>
  </costs>
  <label>eth0</label>
</network_interface>
</network_interfaces>
<smart_servers type="array">
<smart_server>
  <id type="integer">1214</id>
  <costs type="array">
    <cost>
      <value type="integer">6</value>
      <cost type="float">0.0</cost>
      <resource_name>template</resource_name>
    </cost>
    <cost>
      <value type="integer">0</value>
      <cost type="float">0.0</cost>
      <resource_name>cpu_usage</resource_name>
    </cost>
  </costs>
  <label>OH-site</label>
</smart_server>
</smart_servers>
</billing_stats>
<total_cost type="float">0.0</total_cost>
<vm_resources_cost type="float">0.0</vm_resources_cost>
<usage_cost type="float">0.0</usage_cost>
</vm_stats>
Where:

- `created_at` – the timestamp in DB when this record was created
- `updated_at` – the time stamp in DB when this record was updated
- `currency_code` - currency in which this virtual server is charged within the bucket
- `id` – the ID of the server hourly statistics. You can add this parameter to the request URL to get a shorter statistics output.
- `stat_time` – the particular hour for which these statistics were generated
- `user_id` - the ID of VS owner
- `virtual_machine_id` - ID of a smart server
- `virtual_machine_billing_statistics_id` -ID of a smart server billing statistics

`billing_stats` - an array of billing details for the resources used by this smart server:

- `disks` - an array of disks used by this smart server with their billing details:
  - `label` - disk name used in UI
  - `id` - disk ID used in database
  - `costs` - an array of disk related resources with their total prices for the period specified in the stat-time parameter, where:
    - `resource_name` - the resource in question. This can be `disk_size`, `data_read`, `data_written`, `reads_completed` and `writes_completed`
    - `value` - the amount of resources used (GBs of disk size, Kbs of data read/written, the number of reads/writes)
    - `cost` - the total due for the resource

- `network_interfaces` - an array of network interfaces used by this VS with their billing statistics:
  - `label` - network interface name used in OnApp
  - `id` - network interface ID
  - `costs` - an array of network interface related resources with their total prices for the period specified in the stat-time parameter, where:
    - `resource_name` - the resource in question. This can be `ip_addresses`, `rate`, `data_received` and `data_sent`
    - `value` - the amount of resources used by this network interface (the number of IPs, the port speed in Mb per second, the data sent and received in GB )
    - `cost` - the total due for the resource

- `virtual_machines` - an array of virtual server billing details:
  - `label` - VS name
  - `costs` - An array of VS resources with their total prices for the period specified in the stat-time parameter, where:
    - `resource_name` - the resource in question. This can be `cpu_shares`, `cpus`, `memory`, `cpu_usage` and `template`
    - `value` - the amount of resources allocated to this VS. For the templates resource, this parameter means a template ID in database.
    - `cost` - the total due for this resource
  - `id` - virtual server ID
- **total_cost** – the total amount of money owed for the VS specified by id parameter for a particular hour specified by stat_time parameter (total_cost = vm_resources_cost + usage_cost)
- **vm_resources_cost** – the amount of money due for the VS resources for the particular hour specified by stat_time parameter (memory, disks, templates)
- **usage_cost** – the total due for VS usage for this particular hour specified by stat_time parameter (data sent/received, bandwidth, CPU usage)

### 71.21 Search Smart Servers by Label

To search smart servers by label, use the following request:

GET /smart_servers.xml?q=label
GET /smart_servers.xml?q=label

**XML Request Example**

```
```

**JSON Request Example**

```
```

Specify the label of a smart server you are searching for.

### 71.22 Get Smart Server CPU Usage Statistics

To view CPU usage statistics of a virtual server, use the following request:

GET /smart_servers/:smart_server_id/cpu_usage.xml
GET /smart_servers/:smart_server_id/cpu_usage.json

Define a shorter period by setting Start and End time in the API call:


**XML Request Example**
XML Request Example

```bash
curl -i -X GET -u user:userpass --url
http://onapp.test/smart_servers/12/cpu_usage.xml
```

Specify the smart server ID.

Use the following formula to convert CPU data received in the API output:

\[
CPU = \frac{cpu\_time}{10} / 3600
\]

Where \(cpu\_time\) is data from API output.

For example: \(cpu\_time = 2330\), then: \(\frac{2330}{10}/3600=0.06\) (6%).

We use "\(cpu\_time\) * 10" to correct store fractional values.

71.23 Resize Smart Server

To resize a smart server, use the following request:

POST /smart_servers/:smart_server_id/resize.xml
POST /smart_servers/:smart_server_id/resize.json

XML Request Example

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '=?xml version="1.0" encoding="UTF-8"?>smart_server<memory>512</memory><cpus>2</cpus><cpu_shares>30</cpu_shares><allow_cold_resize>1</allow_cold_resize></smart_server>' --url
http://onapp.test/smart_servers/12/resize.xml
```

JSON Request Example

```bash
```

Where:

- `memory` - the amount of RAM allocated to your Smart Server in MB
- `cpus` - the number of CPUs
- `cpu_shares` - CPU priority in %
- `allow_cold_resize` - set 1 to switch to cold resize when hot resize failed
You can also resize a Smart Server using the PUT method (see Edit Smart Server section).

### 71.24 Add/Edit Admin/User Note for Smart Servers

To edit/make an admin note, use the following request:

```plaintext
PUT /smart_servers/:smart_server_id.xml
PUT /smart_servers/:smart_server_id.json
```

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/smart_servers/12.xml -d
  '<smart_server><admin_note>agfagwe tiuuytjgh yuytu</admin_note></smart_server>' -H
  'Accept:application/xml' -H 'Content-type:application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/smart_servers/12.json -d
  '{"smart_server":{"admin_note":"kjfjhjtrtjt"}}' -H
  'Accept:application/json' -H 'Content-type:application/json'
```

Where:

- **admin_note** – enter the text of your note.
- **smart_server_id** - the ID of the smart server to which you add/edit a note

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/smart_servers/12.xml -d
  '<smart_server><note>agfagwe tiuuytjgh yuytu</note></smart_server>' -H
  'Accept:application/xml' -H 'Content-type:application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/smart_servers/12.json -d
  '{"smart_server":{"note":"kjfjhjtrtjt"}}' -H
  'Accept:application/json' -H 'Content-type:application/json'
```

Where:

- **note** – enter the text of your note.
- **smart_server_id** - the ID of the smart server to which you add/edit a note

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no smart server with a requested ID, or URL is incorrect.

### 71.25 Get List of Smart Server Blacklisted Domains

To view the list of smart server blacklisted domains, use the following request:
GET /smart_servers/:smart_server_id/blacklist_domains.xml
GET /smart_servers/:smart_server_id/blacklist_domains.json

XML Request Example

curl -i -X GET
http://onapp.test/smart_servers/xungcyuakcyyeb/blacklist_domains.xml -u
user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X GET
http://onapp.test/smart_servers/xungcyuakcyyeb/blacklist_domains.json -u
user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

XML Output Example

<hash>
  <blacklist_domains type="array">
    <blacklist_domain>site4.com</blacklist_domain>
    <blacklist_domain>site5.com</blacklist_domain>
  </blacklist_domains>
</hash>

Where:
blacklist_domain - the label of the domain to be blacklisted from being accelerated

71.26 Edit Smart Server Blacklisted Domains

To edit blacklisted domains, use the following request:
PUT /smart_servers/:smart_server_id/blacklist_domains.xml
PUT /smart_servers/:smart_server_id/blacklist_domains.json

XML Request Example

curl -i -X PUT
http://onapp.test/smart_servers/xungcyuakcyyeb/blacklist_domains.xml -d
  '<blacklist_domains><hostname_blacklists
type="array"><hostname_blacklist>site4.com</hostname_blacklist></hostname_ 
blacklists></blacklist_domains>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example
curl -i -X PUT
http://onapp.test/smart_servers/xungcyakcyeb/blacklist_domains -d
'{"blacklist_domains":{"hostname_blacklists":["site4.com"]}}' -u
user:userpass -H 'Accept: application/json' -H 'Content-type:
application/json'

**Where:**

*hostname_blacklist* - the label of the domain to be blacklisted from being accelerated

### 71.27 Remove All Smart Server Domains from Blacklist

To remove all IP addresses from a blacklist, use the following request:

**XML Request Example**

```bash
curl -i -X PUT
http://onapp.test/smart_servers/xungcyakcyeb/blacklist_domains.xml -d
'<blacklistDomains><hostnameBlacklists type="string"></hostnameBlacklists></blacklistDomains>' -u user:userpass
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT
http://onapp.test/smart_servers/xungcyakcyeb/blacklist_domains -d
'{"blacklist_domains":{"hostname_blacklists":[]}}' -u user:userpass -H
'Accept: application/json' -H 'Content-type: application/json'
```
72 Software Licenses

When you create a virtual server from a template based on a licensed Operating System, or other licensed software, you need to add a valid license to the system. Use the software_licenses API class to manage licenses. All methods are available to this class.

72.1 Get List of Software Licenses

To get the list of available software licenses, use the following request:

GET /software_licenses.xml
GET /software_licenses.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<software_licenses type="array">
<software_license>
<created_at type="datetime">2011-02-18T01:34:33+02:00</created_at>
<updated_at type="datetime">2011-03-16T00:31:08+02:00</updated_at>
<license>TZXTC-R4GGG-9TT3V-DYDY4-T628B</license>
<total type="integer">20</total>
<arch>x64</arch>
<id type="integer">3</id>
<distro>2008</distro>
<count type="integer">7</count>
<tail> </tail>
<edition>ENT</edition>
</software_license>
</software_licenses>
```

Where:

- **created_at** – the date when the record in DB was created
- **updated_at** - the date when the record in DB was updated
- **license** – the license for the software on which the template will be based
- **total** – the total number of machines allowed by the license
- **arch** – Windows OS architecture (x64 or x86)
- **id** – the ID of the record
- **count** – the number of licenses used of a total allowed
tail – parameter specifies the updated release of Windows OS distribution. If updated, than parameter is R2, otherwise – empty.

edition – Windows OS edition or an array of editions if allowed by the license (STD – Standard, ENT – Enterprise, WEB – web and DC – Data center)

72.2 Get Software License Details

To get details for a particular software license, use the following request:

GET /software_licenses/:id.xml
GET /software_licenses/:id.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<software_license>
  <created_at type="datetime">2011-03-01T12:42:03+02:00</created_at>
  <updated_at type="datetime">2011-03-08T13:54:17+02:00</updated_at>
  <license>TTXTC-R6FFF-9FF3V-DYDY4-T628B</license>
  <total type="integer">100</total>
  <arch>x86</arch>
  <id type="integer">11</id>
  <distro>2003</distro>
  <count type="integer">2</count>
  <tail></tail>
  <edition type="array">
    <string>STD</string>
  </edition>
</software_license>
```

Where:

created_at – the date when the record in DB was created

updated_at - the date when the record in DB was updated

license – the license for the software on which the template will be based

total – the total number of machines allowed by the license

arch – Windows OS architecture (x64 or x86)

id – the ID of the record


count – the number of licenses used of a total allowed
tail – parameter specifies the updated release of Windows OS distribution. If updated, than parameter is R2, otherwise – empty.

edition – Windows OS edition or an array of editions if allowed by the license (STD – Standard, ENT – Enterprise, WEB – web and DC – Data center)

72.3 Add Software License

To add a software license, use the following request:

**POST /software_licenses.xml**
**POST /software_licenses.json**

**XML Request Example**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '
' --url
http://onapp.test/software_licenses.xml
```

**Where:**

arch* - Windows OS architecture (x64 or x86)
total* - the total number of machines allowed by the license
count* - the number of licenses used of a total allowed
tail* - parameter specifies the updated release of Windows OS distribution. If updated, than parameter is R2, otherwise – empty
edition* - Windows OS edition or an array of editions if allowed by the license (STD – Standard, ENT – Enterprise, WEB – web and DC – Data center)
license* - the license for the software on which the template will be based

72.4 Edit Software License

To edit a software license details, use the following request:

**PUT /software_licenses/:id.xml**
**PUT /software_licenses/:id.json**

**XML Request Example**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '
' --url
http://onapp.test/software_licenses/12.xml
```

**Where:**

arch - Windows OS architecture (x64 or x86)
**total** - the total number of machines allowed by the license


**count** - the number of licenses used of a total allowed

**tail** - parameter specifies the updated release of Windows OS distribution. If updated, than parameter is R2, otherwise – empty

**edition** - Windows OS edition or an array of editions if allowed by the license (STD – Standard, ENT – Enterprise, WEB – web and DC – Data center)

**license** - the license for the software on which the template will be based

### 72.5 Delete Software License

To delete a software license, use the following request:

DELETE /software_licenses/:id.xml
DELETE /software_licenses/:id.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/software_licenses/12.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/software_licenses/12.json
```
73 SSH keys

This chapter describes the process of adding and managing SSH keys for a user profile.

73.1 Get List of SSH Keys

To see all the SSH keys in the cloud, use the following request:

GET /settings/ssh_keys.xml
GET /settings/ssh_keys.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example:

```
<ssh_keys type="array">
  <ssh_key>
    <created_at type="datetime">2011-09-13T16:10:02Z</created_at>
    <updated_at type="datetime">2011-09-13T16:10:02Z</updated_at>
    <id type="integer">3</id>
    <user_id type="integer">1</user_id>
    <key>ssh-rsa
        AAAAB3NzaC1yc2EAAAABJQAAIAIEAqzsLk+oPP9Qxz0Xgpqoe9DqNV7Qe3+oig/o6Ub70Yh+2a 
        rf8NxtqeqamC1Kr1Lm120Awd38d2OCW64zuNjhsS1SH6 
        gSjJUYWHsi7jUBl0v1GtJ7jsWdhgaKkjk1vXH3YFLTHPuKU+pc=user@onapp.test
    </key>
  </ssh_key>
</ssh_keys>
```

Where:

- `ssh_key` – an array which displays the key info
- `id` – the SSH key ID
- `user_id` – ID of the user to whom the key belongs
- `key` – SSH key

73.2 Add SSH Key

To add SSH keys to a user profile, use the following request:

POST /users/:user_id/ssh_keys.xml
POST /users/:user_id/ssh_keys.json
XML Request Example

```
curl -X POST -u user:userpass http://onapp.test/users/12/ssh_keys.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d"'<ssh_key><key> ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIEAgqqlz0Xgpqoe9DqNV7Qe3+oig/o6Ub30Yh+2a rf8Nxtqeq1m12t12d0Aw38d2OUC6Erw/2ciwzz2Ib0MLrTyjLNe2CW64uNhS1SH6 gSjJUYwHS17jUB10v1GtJ7jswBdhaKkjk1vXH3YFLTHPuKU+pc= user@onapp.test</key></ssh_key>'"
```

JSON Request Example

```
curl -X POST -u user:userpass http://onapp.test/users/12/ssh_keys.json -H 'Accept: application/json' -H 'Content-type: application/json' -d'"{"ssh_key":{"key":"ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIEAgqqlz0Xgpqoe9DqNV7Qe3+oig/o6Ub30Yh+2a rf8Nxtqeq1m12t12d0Aw38d2OUC6Erw/2ciwzz2Ib0MLrTyjLNe2CW64uNhS1SH6 gSjJUYwHS17jUB10v1GtJ7jswBdhaKkjk1vXH3YFLTHPuKU+pc= user@onapp.test"}}"'
```

Where:

- `key` - a SSH key in the following format: `ssh-[type] [ascii-symbols allowed for base64 string] [user credentials]`

73.3 Edit SSH Key

To edit an SSH key you may use both types of the following requests:

PUT `/users/:user_id/ssh_keys/:id.xml`

PUT `/users/:user_id/ssh_keys/:id.json`

or

PUT `/settings/ssh_keys/:id.xml`

PUT `/settings/ssh_keys/:id.json`

XML Request Example

```
curl -X POST -u user:userpass http://onapp.test/users/12/ssh_keys/3.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d"'<ssh_key><key> ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIEAgqqlz0Xgpqoe9DqNV7Qe3+oig/o6Ub30Yh+2a rf8Nxtqeq1m12t12d0Aw38d2OUC6Erw/2ciwzz2Ib0MLrTyjLNe2CW64uNhS1SH6 gSjJUYwHS17jUB10v1GtJ7jswBdhaKkjk1vXH3YFLTHPuKU+pc= user@onapp.test</key></ssh_key>'"
```

JSON Request Example

```
curl -X POST -u user:userpass http://onapp.test/users/12/ssh_keys/3.json -H 'Accept: application/json' -H 'Content-type: application/json' -d'"{"ssh_key":{"key":"ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIEAgqqlz0Xgpqoe9DqNV7Qe3+oig/o6Ub30Yh+2a rf8Nxtqeq1m12t12d0Aw38d2OUC6Erw/2ciwzz2Ib0MLrTyjLNe2CW64uNhS1SH6 gSjJUYwHS17jUB10v1GtJ7jswBdhaKkjk1vXH3YFLTHPuKU+pc= user@onapp.test"}}"'
```
73.4 Delete SSH Key

To delete an SSH from the system (and from the user profile), use the following request:

DELETE /settings/ssh_keys/:id.xml
DELETE /settings/ssh_keys/:id.json

**XML Request Example**

```
curl -X DELETE -u user:userpass http://onapp.test/settings/ssh_keys/12.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -X DELETE -u user:userpass http://onapp.test/settings/ssh_keys/12.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```
74 Statistics

The Statistics section unites usage trends and cloud usage statistics.

74.1 Usage Trends Statistics

To view usage trends statistics, use the following request:

GET /billing/dashboard_statistics.xml
GET /billing/dashboard_statistics.json

XML Request Example:

curl -X GET

Where:

stats_for - leave this parameter empty to get all statistics data. To get a specific data, indicate any of these values: cpus, memory, disk_size, virtual_servers, smart_servers, baremetal_servers.

period - indicate start date and end date for the period, for which you want to get the statistics.

JSON Request Example:


Where:

- [1466568000000, 29] - time (in milli seconds), value array.

If period <= 24 hours the API response contains hourly statistics, else (> 24 hours) - returns daily statistics.

XML Output Example
74.2 Cloud Usage Statistics

Cloud Usage statistics show detailed information on the resources used by virtual servers. To get the daily stats (information on the resources used by virtual servers), use the following request:

GET /usage_statistics.xml
GET /usage_statistics.json

This request sends back usage statistics for all virtual servers in the cloud (per VS for the last 48 hours).

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<vm_stats>
<vm_stat>
<data_sent>0.0</data_sent>
<reads_completed>328892.0</reads_completed>
<data_received>0.0</data_received>
<cpu_usage>2813.0</cpu_usage>
<virtual_machine_id>883</virtual_machine_id>
<data_read>1315568.0</data_read>
<writes_completed>193395.0</writes_completed>
<user_id>1</user_id>
<data_written>773580.0</data_written>
</vm_stat>
...
<vm_stat></vm_stat>
...
</vm_stats>

Where:

data_sent - the amount of Kilobytes sent by this VS
reads_completed - the number of read operations performed by the disk
data_received - the amount of Kilobytes received by this VS
cpu_usage - the average CPU percentage that the VS has been using during the last 72 hours or during the specified period.
virtual_machine_id - the ID of the VS for which these statistics are generated
writes_completed - the number of write operations performed by the disk
data_read - the amount of data read from a disk in Kilobytes
data_written - the amount of data written to a disk in Kilobytes

Other statistics generated in the system:

- Get User Statistics
- View Billing Statistics for User
- View Disk IOPS (Input/Output Statistics)
- Billing Statistics for VS
- Billing Statistics for CDN Edge Servers
- Get Load Balancer Billing Statistics
75 Storage Server Backups

The storage server backup feature enables users to create normal and incremental backups of their storage servers.

75.1 Get All Storage Server Backups Details

To get the details of all server backups, use the following request:

GET /storage_servers/:id/backups.xml
GET /storage_servers/:id/backups.json

XML Request Example

bash

curl -i -X GET -u user:password
http://onapp.test/storage_servers/12/backups.xml

JSON Request Example

bash

curl -i -X GET -u user:password
http://onapp.test/storage_servers/12/backups.xml

XML Output Example
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <backup_size type="integer">495960</backup_size>
    <built type="boolean">true</built>
    <built_at type="datetime">2014-12-24T15:47:25+02:00</built_at>
    <created_at type="datetime">2014-12-24T15:45:31+02:00</created_at>
    <data_store_type type="string">lvm</data_store_type>
    <id type="integer">303</id>
    <identifier>iti18apbz635vr</identifier>
    <initiated>manual</initiated>
    <iqn nil="true"/>
    <locked type="boolean">false</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">20</min_disk_size>
    <min_memory_size type="integer">2048</min_memory_size>
    <note nil="true"/>
    <operating_system type="string">linux</operating_system>
    <operating_system_distro type="string">ubuntu</operating_system_distro>
    <target_id type="integer">8087</target_id>
    <target_type type="string">Disk</target_type>
    <template_id type="integer">2218</template_id>
    <updated_at type="datetime">2014-12-24T15:47:25+02:00</updated_at>
    <user_id type="integer">132</user_id>
    <volume_id nil="true"/>
    <backup_type type="string">normal</backup_type>
    <disk_id type="integer">8087</disk_id>
  </backup>
  ...</backup>
</backups>

Where:

allow_resize_without_reboot - true, if storage server's CPU and RAM can be resized without reboot; a requirement, which will be applicable when the backup is converted into a template

allowed_hot_migrate - true, if hot migration is allowed; a requirement, which will be applicable when the backup is converted into a template

allowed_swap - true, if swap is allowed; a requirement, which will be applicable when the backup is converted into a template

backup_server_id - the ID of the backup server where the backup is stored

backup_size - size of the backup in Kilobytes

built - true, if the storage server backup has been built

data_store_type - data store type: lvm, vmware or SolidFire

id - ID of the backup

identifier - identifier of the backup in the DB

initiated - how the backup was launched - either manually or automatically

iqn - volume ISCSI qualified name; SolidFire - related parameter

locked - true, if the storage server backup is locked due to a running transaction and no new transactions can be initiated at the moment

marked_for_delete - true, if the backup is marked for deletion (for auto-backups)
**min_disk_size** - minimum disk size requirement, which will be applicable when the backup is converted into a template

**min_memory_size** - minimum RAM requirement, which will be applicable when the backup is converted into a template

**note** - an arbitrary note to the backup

**operating_system** - the OS of the storage server backup

**operating_system_distro** - the OS distribution of the storage server backup

**target_id** - ID of a backup target - either a disk or a virtual server

**target_type** - target for which the backup was taken - either a disk or a virtual server

**template_id** - the ID of a template from which the storage server was built

**updated_at** - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

**user_id** - the ID of a user the storage server belongs to

**volume_id** - data store ID

**backup_type** - normal or incremental

---

### 75.2 Get Normal Storage Server Backups Details

To get the details of normal server backups, use the following request:

GET /storage_servers/:id/backups/images.xml
GET /storage_servers/:id/backups/images.json

**XML Request Example**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password --url
http://onapp.test/storage_servers/12/backups/images.xml
```

**JSON Request Example**

```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password --url
http://onapp.test/storage_servers/12/backups/images.json
```

**XML Output example**
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <built type="boolean">true</built>
    <built_at type="datetime">2014-12-24T15:47:25+02:00</built_at>
    <created_at type="datetime">2014-12-24T15:45:31+02:00</created_at>
    <data_store_type>lvm</data_store_type>
    <id type="integer">303</id>
    <identifier>iti18apbz635vr</identifier>
    <initiated>manual</initiated>
    <iqn nil="true"/>
    <locked type="boolean">false</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">20</min_disk_size>
    <min_memory_size type="integer">2048</min_memory_size>
    <note nil="true"/>
    <operating_system>linux</operating_system>
    <operating_system_distro>ubuntu</operating_system_distro>
    <target_id type="integer">8087</target_id>
    <target_type>Disk</target_type>
    <template_id type="integer">2218</template_id>
    <updated_at type="datetime">2014-12-24T15:47:25+02:00</updated_at>
    <user_id type="integer">132</user_id>
    <volume_id nil="true"/>
    <backup_type>normal</backup_type>
    <disk_id type="integer">8087</disk_id>
  </backup>
  ...</backup>
</backups>

Where:

allow_resize_without_reboot - true, if storage server's CPU and RAM can be resized without reboot; a requirement, which will be applicable when the backup is converted into a template

allowed_hot_migrate - true, if hot migration is allowed; a requirement, which will be applicable when the backup is converted into a template

allowed_swap - true, if swap is allowed; a requirement, which will be applicable when the backup is converted into a template

backup_server_id - the ID of the backup server where the backup is stored

backup_size - size of the backup in Kilobytes

built - true, if the storage server backup has been built

data_store_type - data store type: lvm, vmware or SolidFire

id - ID of the backup

identifier - identifier of the backup in the DB

initiated - how the backup was launched - either manually or automatically

iqn - volume ISCSI qualified name; SolidFire - related parameter

locked - true, if the storage server backup is locked due to a running transaction and no new transactions can be initiated at the moment

marked_for_delete - true, if the backup is marked for deletion (for auto-backups)
**min_disk_size** - minimum disk size requirement, which will be applicable when the backup is converted into a template

**min_memory_size** - minimum RAM requirement, which will be applicable when the backup is converted into a template

**note** - an arbitrary note to the backup

**operating_system** - the OS of the storage server backup

**operating_system_distro** - the OS distribution of the storage server backup

**target_id** - ID of a backup target - either a disk or a virtual server

**target_type** - target for which the backup was taken - either a disk or a virtual server

**template_id** - the ID of a template from which the storage server was built

**updated_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

**user_id** - the ID of a user the storage server belongs to

**volume_id** - data store ID

**backup_type** - normal or incremental

### 75.3 Get Incremental Server Backups Details

To get the details of incremental server backups, use the following request:

GET /storage_servers/:id/backups/files.xml
GET /storage-servers/:id/backups/files.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <backup_size type="integer">0</backup_size>
    <built type="boolean">false</built>
    <built_at nil="true"/>
    <created_at type="datetime">2014-12-25T13:32:02+02:00</created_at>
    <data_store_type>lvm</data_store_type>
    <id type="integer">316</id>
    <identifier>bk1ntvx61p32uq</identifier>
    <initiated>manual</initiated>
    <iqn nil="true"/>
    <locked type="boolean">true</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">0</min_disk_size>
    <min_memory_size type="integer">2048</min_memory_size>
    <note>manual BU</note>
    <operating_system>linux</operating_system>
    <operating_system_distro>ubuntu</operating_system_distro>
    <target_id type="integer">7324</target_id>
    <target_type>VirtualMachine</target_type>
    <template_id type="integer">2218</template_id>
    <updated_at type="datetime">2014-12-25T13:32:02+02:00</updated_at>
    <user_id type="integer">132</user_id>
    <volume_id nil="true"/>
    <backup_type>incremental</backup_type>
    <disk_id nil="true"/>
  </backup>
</backups>

Where:

allow_resize_without_reboot - true, if storage server's CPU and RAM can be resized without reboot; a requirement, which will be applicable when the backup is converted into a template

allowed_hot_migrate - true, if hot migration is allowed; a requirement, which will be applicable when the backup is converted into a template

allowed_swap - true, if swap is allowed; a requirement, which will be applicable when the backup is converted into a template

backup_server_id - the ID of the backup server where the backup is stored

backup_size - size of the backup in Kilobytes

built - true, if the storage server backup has been built

data_store_type - data store type: lvm, vmware or SolidFire

id - ID of the backup

identifier - identifier of the backup in the DB

initiated - how the backup was launched - either manually or automatically

iqn - volume ISCSI qualified name; SolidFire - related parameter

locked - true, if the storage server backup is locked due to a running transaction and no new transactions can be initiated at the moment

marked_for_delete - true, if the backup is marked for deletion (for auto-backups)
OnApp Cloud 6.3 API Guide

```
min_disk_size - minimum disk size requirement, which will be applicable when the backup is converted into a template
min_memory_size - minimum RAM requirement, which will be applicable when the backup is converted into a template
note - an arbitrary note to the backup
operating_system - the OS of the storage server backup
operating_system_distro - the OS distribution of the storage server backup
target_id - ID of a backup target - either a disk or a virtual server
target_type - target for which the backup was taken - either a disk or a virtual server
template_id - the ID of a template from which the storage server was built
updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
user_id - the ID of a user the storage server belongs to
volume_id - data store ID
backup_type - normal or incremental
```

### 75.4 Add Backup for Storage Server

To create a backup for a storage server, use the following request:

```
POST /storage_servers/:id/backups.xml
POST /storage_servers/:id/backups.json
```

Depending on the configuration of your cloud, either a normal or an incremental backup for your storage server will be created.

**XML Request Example**

```
```

**JSON Request Example**

```
```

**Where:**

```
note - an arbitrary note to the backup
```

**XML Output Example**
Where:

allow_resize_without_reboot - true, if storage server’s CPU and RAM can be resized without reboot; a requirement, which will be applicable when the backup is converted into a template
allowed_hot_migrate - true, if hot migration is allowed; a requirement, which will be applicable when the backup is converted into a template
allowed_swap - true, if swap is allowed; a requirement, which will be applicable when the backup is converted into a template
backup_server_id - the ID of the backup server where the backup is stored
backup_size - size of the backup in Kilobytes
built - true, if the storage server backup has been built
data_store_type - data store type: lvm, vmware or SolidFire
id - ID of the backup
identifier - identifier of the backup in the DB
initiated - how the backup was launched - either manually or automatically
iqn - volume iSCSI qualified name; SolidFire - related parameter
locked - true, if the storage server backup is locked due to a running transaction and no new transactions can be initiated at the moment
marked_for_delete - true, if the backup is marked for deletion (for auto-backups)
min_disk_size - minimum disk size requirement, which will be applicable when the backup is converted into a template
min_memory_size - minimum RAM requirement, which will be applicable when the backup is converted into a template

note - an arbitrary note to the backup

operating_system - the OS of the storage server backup

operating_system_distro - the OS distribution of the storage server backup

target_id - ID of a backup target - either a disk or a virtual server

target_type - target for which the backup was taken - either a disk or a virtual server

template_id - the ID of a template from which the storage server was built

updated_at - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

user_id - the ID of a user the storage server belongs to

volume_id - data store ID

backup_type - normal or incremental

75.5 Add/Edit Storage Server Backup Note

To update backup with a note, use the following request:

PUT /backups/:backup_id/note.xml
PUT /backups/:backup_id/note.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

Where you have to specify backup ID in the URL.

75.6 Restore Storage Server Backup

To restore a disk from a backup, use the following request:

POST /backups/:backup_id/restore.xml
POST /backups/:backup_id/restore.json

XML Request Example

```bash
curl -X PUT http://onapp.test/storage_servers/13/backups/images/2/restore.xml -d '<restore>'
```

JSON Request Example

76 System Configuration

Lists the configuration settings of your OnApp installation.

76.1 View System Configuration

To see all the system configuration settings, use the following request:

GET /settings/configuration.xml
GET /settings/configuration.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example
<settings>
  <use_yubikey_login type="boolean">true</use_yubikey_login>
  <yubikey_api_key>00000000000000000000</yubikey_api_key>
  <yubikey_api_id>00000</yubikey_api_id>
  <allow_incremental_backups type="boolean">true</allow_incremental_backups>
  <ssh_file_transfer type="boolean">true</ssh_file_transfer>
  <ssh_file_transfer_server>109.123.105.162</ssh_file_transfer_server>
  <ssh_file_transfer_user>root</ssh_file_transfer_user>
  <ssh_file_transfer_options>-o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null
  <ssh_port type="integer">22</ssh_port>
  <template_path>/onapp/templates</template_path>
  <backups_path>/onapp/backups</backups_path>
  <data_path>/onapp/data</data_path>
  <use_yubikey_login type="boolean">true</use_yubikey_login>
  <yubikey_api_key>00000000000000000000</yubikey_api_key>
  <yubikey_api_id>00000</yubikey_api_id>
  <allow_incremental_backups type="boolean">true</allow_incremental_backups>
  <ssh_file_transfer type="boolean">true</ssh_file_transfer>
  <ssh_file_transfer_server>109.123.105.162</ssh_file_transfer_server>
  <ssh_file_transfer_user>root</ssh_file_transfer_user>
  <ssh_file_transfer_options>-o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null
  <ssh_port type="integer">22</ssh_port>
  <template_path>/onapp/templates</template_path>
  <backups_path>/onapp/backups</backups_path>
  <data_path>/onapp/data</data_path>
  <delete_template_source_after_install type="boolean">true</delete_template_source_after_install>
  <license_key>FD1E0-4766C-C69E9-18A83-8CB60-1DB62</license_key>
  <generate_comment># Automatically generated by OnApp (3.1.0)</generate_comment>
  <simultaneous_backups type="integer">1</simultaneous_backups>
  <simultaneous_backups_per_datastore type="integer">150</simultaneous_backups_per_datastore>
  <simultaneous_backups_per_hypervisor type="integer">2</simultaneous_backups_per_hypervisor>
  <simultaneous_transactions type="integer">10</simultaneous_transactions>
  <guest_wait_time_before_destroy type="integer">300</guest_wait_time_before_destroy>
  <remote_access_session_start_port type="integer">30000</remote_access_session_start_port>
  <remote_access_session_last_port type="integer">30099</remote_access_session_last_port>
  <ajax_power_update_time type="integer">10000</ajax_power_update_time>
  <ajax_pagination_update_time type="integer">10000</ajax_pagination_update_time>
  <hypervisor live times type="integer">12</hypervisor live times>
  <recovery_templates_path>/onapp/tools/recovery</recovery_templates_path>
  <remove_backups_on_destroy_vm type="boolean">true</remove_backups_on_destroy_vm>
  <disable_hypervisor_failover type="boolean">false</disable_hypervisor_failover>
  <disable_billing type="boolean">true</disableBilling>
  <ips_allowed_for_login type="array">
    <ips_allowed_for_login>1.2.3.4</ips_allowed_for_login>
    <ips_allowed_for_login>5.6.7.8</ips_allowed_for_login>
  </ips_allowed_for_login>
  <monitis_path>/usr/local/monitis</monitis_path>
  <monitis_account>monitis@onapp.com</monitis_account>
  <monitis_apikey>4JIA5DJUM9001HJ550B88V73GR</monitis_apikey>
  <locales type="array">
    <locale>en</locale>
    <locale>ja</locale>
  </locales>
  <max_memory_ratio type="integer">16</max_memory_ratio>
  <remove_old_root_passwords type="boolean">false</remove_old_root_passwords>
  <pagination_max_items_limit type="integer">99</pagination_max_items_limit>
  <default_image_template type="integer">19</default_image_template>
  <service_account_name>onapp</service_account_name>
  <default_acceleration_policy type="boolean">false</default_acceleration_policy>
  <default_virsh_console_policy type="boolean">false</default_virsh_console_policy>
  <default_firewall_policy type="boolean">DROP</default_firewall_policy>
<drop_firewall_policy_allowed_ips>69.165.230.31, 10.81.0.11</drop_firewall_policy_allowed_ips>
<app_name>DEV5 cloud</app_name>
<show_ip_address_selection_for_new_vm type="boolean">true</show_ip_address_selection_for_new_vm>
<transaction_approvals>true</transaction_approvals>
<backup_taker_delay type="integer">5</backup_taker_delay>
<billing_stat_updater_delay type="integer">5</billing_stat_updater_delay>
<cluster_monitor_delay type="integer">15</cluster_monitor_delay>
<hypervisor_monitor_delay type="integer">5</hypervisor_monitor_delay>
<cdn_sync_delay type="integer">1200</cdn_sync_delay>
<google_map_token/>
<dashboard_stats type="array">
<dashboard_stat>cpu</dashboard_stat>
<dashboard_stat>memory</dashboard_stat>
</dashboard_stats>
<schedule_runner_delay type="integer">5</schedule_runner_delay>
<transaction_runner_delay type="integer">5</transaction_runner_delay>
<zombie_transaction_time type="integer">180</zombie_transaction_time>
<zombie_disk_space_updater_delay type="integer">300</zombie_disk_space_updater_delay>
<run_recipe_on_vs_sleep_seconds type="integer">10</run_recipe_on_vs_sleep_seconds>
<dns_enabled type="boolean">true</dns_enabled>
<enabled_libvirtAntiSpoofing type="boolean">true</enabled_libvirtAntiSpoofing>
<allow_start_vms_with_one_ip type="boolean">true</allow_start_vms_with_one_ip>
<allow_initial_root_password_encryption type="boolean">true</allow_initial_root_password_encryption>
<wipe_out_disk_on_destroy type="boolean">false</wipe_out_disk_on_destroy>
<partition_align_offset type="integer">2048</partition_align_offset>
<password_enforce_complexity type="boolean">false</password_enforce_complexity>
<password_minimum_length type="integer">6</password_minimum_length>
<password_upper_lowercase type="boolean">true</password_upper_lowercase>
<password_letters_numbers type="boolean">true</password_letters_numbers>
<password_symbols type="boolean">true</password_symbols>
<password_force_unique type="boolean">true</password_force_unique>
<password_lockout_attempts type="integer">1</password_lockout_attempts>
<password_expiry type="integer">1</password_expiry>
<password_history_length type="integer">12</password_history_length>
<force_windows_backups type="boolean">false</force_windows_backups>
<cloud_boot_enabled type="boolean">true</cloud_boot_enabled>
<nfs_root_ip>onapp.test</nfs_root_ip>
<cloud_boot_target>onapp.test</cloud_boot_target>
<max_ip_addresses_to_assign_simultaneously type="integer">256</max_ip_addresses_to_assign_simultaneously>
<storage_enabled type="boolean">true</storage_enabled>
<intra_hypervisor_balance_threshold_ratio type="integer">95</intra_hypervisor_balance_threshold_ratio>
<inter_hypervisor_balance_threshold_ratio type="integer">95</inter_hypervisor_balance_threshold_ratio>
<uniform_node_capacity_threshold_ratio type="integer">5</uniform_node_capacity_threshold_ratio>
<prefer_local_reads type="boolean">false</prefer_local_reads>
<vmware_cluster_name>OnApp</vmware_cluster_name>
<ip_address_reservation_time type="integer">60</ip_address_reservation_time>
<allow_hypervisor_password_encryption type="boolean">false</allow_hypervisor_password_encryption>
<archive_stats_period type="integer">3</archive_stats_period>
<instant_stats_period type="integer">10</instant_stats_period>
<brarchive_stats_enabled type="boolean">true</brarchive_stats_enabled>
<brsystem_alert_reminder_period type="integer">50</brsystem_alert_reminder_period>
<bruse_html5_vnc_console type="boolean">true</bruse_html5_vnc_console>
<brmax_network_interface_port_speed type="integer">10000</brmax_network_interface_port_speed>
<brsession_timeout type="integer">480</brsession_timeout>
<enable_super_admin_permissions type="boolean">false</enable_super_admin_permissions>
<brurl_for_custom_tools type="string"/>
<brbackup_convert_coefficient type="float">1.1</backup_convert_coefficient>
<brrsync_option_xattrs type="boolean">false</rsync_option_xattrs>
<brrsync_option_acls type="boolean">true</rsync_option_acls>
<brsimultaneous_backups_per_backup_server type="integer">3</brsimultaneous_backups_per_backup_server>
<brenable_hourly_storage_report type="boolean">false</brenable_hourly_storage_report>
<brenable_daily_storage_report type="boolean">false</brenable_daily_storage_report>
<brstorage_unicast type="boolean">true</storage_unicast>
<brsnmptrap_addresses type="string">onapp.test</brsnmptrap_addresses>
<brsnmptrap_port type="integer">3162</brsnmptrap_port>
<brinfiband_cloud_boot_enabled type="boolean">false</brinfiband_cloud_boot_enabled>
<brcdn_max_results_per_get_page type="integer">500</brcdn_max_results_per_get_page>
<brinstance_packages_threshold_num type="integer">6</brinstance_packages_threshold_num>
<brallow_to_collect_errors type="boolean">true</allow_to_collect_errors>
<brdraas_enabled type="boolean">false</draas_enabled>
<brzabbix_host type="string">000.000.00.00</brzabbix_host>
<brzabbix_url type="string">/zabbix</brzabbix_url>
<brzabbix_user type="string">Admin</brzabbix_user>
<brzabbix_password type="string">zabbix</brzabbix_password>
<brlicence_key type="string">FD1E0-4766C-C69E9-19A83-8CB60-1DE62</brlicence_key>
<brsystem_theme type="string">dark</brsystem_theme>
<brrecipe_temporary_directory type="string">/tmp</brrecipe_temporary_directory>
<brmigration_rate_limit type="integer">10</migration_rate_limit>
<brsimultaneous_migrations_per_hypervisor type="integer">5</brsimultaneous_migrations_per_hypervisor>
<brisolated_license type="boolean">true</isolated_license>
<brpagination_dashboard_pages_limit type="integer">10</brpagination_dashboard_pages_limit>
<brallow_advanced_vs_management type="boolean">true</allow_advanced_vs_management>
</settings>

Where:

use_yubikey_login - whether logging in using a Yubikey is enabled

yubikey_api_key - your Yubico secret key

yubikey_api_id - your Yubico client ID

use_ssh_file_transfer - set 1 to allow secure file access, transfer and management to a remote server. It is not possible to utilize SSH file transfer option when incremental backups are enabled.

Skip this option if you are using incremental backups.
If you are using incremental backups option, you should either enable dedicated backup servers in your cloud or share the backups and templates folders (paths) between your compute resources.

- **ssh_file_transfer_server** - the address of the remote server
- **ssh_file_transfer_user** - the login used for remote server authentication. A password is not required, as it is required that you store a host key
- **ssh_file_transfer_options** - SSH protocol options that set the rules and behavior of how to log into the remote server
- **ssh_port** - the port used for contacting SSH servers
- **template_path** - path to the directory where templates will be stored
- **backups_path** - path to the directory where backups will be stored
- **data_path** - path to the directory where the recovery templates will be stored
- **update_server_url** - URL address where OnApp software updates are downloaded from
- **delete_template_source_after_install** - if enabled, the downloaded templates will be deleted after they are distributed
- **license_key** - license key of your OnApp CP
- **generate_comment** - this text is added by OnApp to system configuration files, such as resolv.conf
- **simultaneous_backups** - the maximum allowed number of simultaneous compute resource and data store backup processes
- **simultaneous_backups_per_datastore** - the maximum number of simultaneous data store backup processes
- **simultaneous_backups_per_hypervisor** - the maximum number of simultaneous compute resource backup processes
- **simultaneous_transactions** - the number of transaction runners which the daemon will execute at the same time
- **guest_wait_time_before_destroy** – the VS shutdown period (from 30 to 300 seconds). This allows to refuse the shutdown if the VS is booting and retry every 30 seconds till it is registered.
- **remote_access_session_start_port** - the first port in the range, which are used to remotely connect to virtual servers using the integrated VNC console
- **remote_access_session_last_port** - the last port in the range, which are used to remotely connect to virtual servers using the integrated VNC console
- **support_help_email** - support email to which the help requests will be sent from Control Panel > Help.
- **ajax_power_update_time** - how often VS status is refreshed on the Virtual Servers screen in ms
- **ajax_pagination_update_time** - how often the dashboard, logs and other items are refreshed in ms
- **hypervisor_live_times** - determines how many times the Control Panel server will attempt to contact a compute resource before failover is initiated. 1 compute resource live time = 10 seconds
- **recovery_templates_path** - path to the directory where recovery templates will be stored
- **remove_backups_on_destroy_vm** - set 1 to remove all VS backups after this VS was deleted
disable_hypervisor_failover - 1, if compute resource failover will not initiate after meeting the value of the hypervisor_live_times parameter

disable_billing - true, if billing is disabled, otherwise false

ips_allowed_for_login - list of IP addresses allowed for login to OnApp CP

Note that Monitis will come to its end of life on June 1st, 2019 and will no longer be supported.

monitis_path - path to the directory where Monitis client (to enable autoscale) will be installed

monitis_account - name of the Monitis account

monitis_apikey - API key to access the Monitis account

locales - an array of locals (the locale code) available for the users

remove_old_root_passwords - 1, if old root password is removed, otherwise 0

pagination_max_items_limit – the maximum number of items after which the Show All option cannot be applied.

pagination_dashboard_pages_limit – the maximum number of pages to list log items in the Activity Log section at the main Dashboard page.

allow_advanced_vs_management - true, if the VS advanced configuration is enabled, otherwise, false

default_image_template - default VS template to create a new virtual server

service_account_name - service account name that is automatically created on VMware virtual servers to be able to communicate with them.

default_acceleration_policy - true, if default acceleration policy is enabled, otherwise, false

_default_virsh_console_policy - true, if Virsh console is available by default for all newly created VSs, otherwise, false

default_firewall_policy - default firewall policy for all VSs (unless set otherwise for a particular VS)

drop_firewall_policy_allowed_ips - enter the IP addresses to be allowed as an exception if the default firewall policy is DROP

app_name - application name displayed on the login screen

show_ip_address_selection_for_new_vm - set 1 to enable IP address assignment during VS creation

transaction_approvals - true, if transaction approvals are enabled, otherwise, false

backup_taker_delay - frequency in seconds for launching the Backup Taker task

billing_stat_updater_delay - frequency in seconds for launching the Billing Stats Monitor task

calendar_monitor_delay - frequency in seconds for launching the Calendar Monitor task

hypervisor_monitor_delay - frequency in seconds for launching the compute resource Monitor task

cdn_sync_delay - frequency in seconds for launching the CDN sync task

billing_stat_updater_delay - frequency in seconds for launching the Billing Stats Monitor task

cluster_monitor_delay - frequency in seconds for launching the Cluster Monitor task

hypervisor_monitor_delay - frequency in seconds for launching the compute resource Monitor task

cdn_sync_delay - frequency in seconds for launching the CDN sync task

google_map_token - add Google API key, if you face the problem with viewing the maps on VS/Smart/Application server creation wizard.

dashboard_stats - an array of statistics, which is shown on the dashboard
schedule_runner_delay - frequency in seconds for launching the Schedule Runner task
transaction_runner_delay - frequency in seconds for launching the Transaction Runner task
zombie_transaction_time – the duration in minutes for the system to consider a task as “pending” before it can be relegated to zombie status
zombie_disk_space_updater_delay – the duration in minutes for the system to check the size of a zombie disk
dns_enabled – 1, if DNS is enabled, otherwise 0
enabled_libvirt AntiSpoofing - 1, if the anti-spoofing protection is enabled, otherwise 0
allow_start_vms_with_one_ip - true, if it is allowed to start up virtual servers with one IP address, otherwise false
allow_initial_root_password_encryption - set 1 to allow password encryption, otherwise set 0
allow_hypervisor_password_encryption - set 1 to allow VMware compute resource password encryption, otherwise set 0
wipe_out_disk_on_destroy – 1, if the disk wipeout is allowed when deleting or migrating a disk, otherwise 0
password_minimum_length type - the minimum required password length
password_letters_numbers - 1, if the user is enforced to use both letters and numbers in their password, otherwise 0
password_symbols type - 1, if the user is enforced to use symbols in their password, otherwise 0
password_force_unique - 1, if the user is enforced to enter unique password configuration each time they change the password, otherwise 0. This refers to the user account passwords only.
password_lockout_attempts - the number of unsuccessful logon attempts that are allowed before user’s account is locked out
password_lockout_response - the message displayed to the user after they exceed the number of unsuccessful logon attempts
password_expiry type - password expiry period in months
password_history_length - the number of last passwords saved in OnApp configuration
force_windows_backups - 1, if forced backup feature is enabled for Windows-based virtual servers, otherwise 0
cloud_boot_enabled - 1, if CloudBoot system is enabled on the cloud, otherwise 0
nfs_root_ip - IP address of the NFS server where templates for creating PXE compute resources are stored
cloud_boot_target - IP of the server where the CP is installed
max_ip_addresses_to_assign_simultaneously - the maximum number of IP addresses that can be assigned to user simultaneously. The default value is 256.
storage_enabled - true, if OnApp storage is enabled, otherwise false
intra_hypervisor_balance_threshold_ratio - percentage ratio to the average free space for all nodes in the hypervisor. The default value is 5%. You will receive a warning in the storage
health check if the free space on one of the compute resource's nodes will be smaller by the indicated value as compared to the other nodes on the compute resource.

`inter_hypervisor_balance_threshold_ratio` - percentage ratio to the average free space for all compute resources within one hypervisor zone. The default value is 5%. You will receive a warning in the storage health check if the free space on one of the compute resources in the zone will be smaller by the indicated value as compared to the other resources in the compute zone.

`uniform_node_capacity_threshold_ratio` - the value (%) by which the size of a node in a data store can differ from the average node size in this data store. The default value is 5%. If this value is exceeded, you will receive a warning in the storage health check.

`prefer_local_reads` - true, if the local read path is enabled, otherwise false

`vmware_cluster_name` - name of the VMware cluster

`ip_address_reservation_time` - the duration in seconds during which the IP address will be reserved for a user and unavailable for other users. The default value is 60.

`license_key` - license key of your OnApp CP

`isolated_license` - true, if the isolated license is used on the CP, otherwise false

`archive_stats_period` - hourly statistics storage time set in months. For example, if you set that parameter to 10, the hourly statistics will be stored for the current month and the 10 previous months. And everything older than 10 months will be sent to archive (that is converted into monthly statistics). If this parameter is set as 1, then you can view the detailed hourly statistics for both the current and the previous month.

`instant_stats_period` - the number of days the instant (raw) statistics will be stored

`is_archive_stats_enabled` - true, if the hourly statistics archiving is enabled, otherwise false

`use_html5_vnc_console` - true, if the use of HTML 5 console is enabled, otherwise false

---

NOTE: It is only possible to use HTML 5 console if the Control Panel server is based on CentOS 6.

---

`backup_convert_coefficient` - the parameter is applicable only to incremental backups. During the backup conversion to template, backup’s size is multiplied by this coefficient to make sure that template will be slightly bigger than the actual size for correct performance.

`session_timeout` - the timeout between sessions within OnApp in minutes

💡 `enable_super_admin_permissions` - true, if a super admin feature is enabled; otherwise, false


`snmptrap_addresses*` - a set of IPv4 addresses separated by coma. These IP addresses will be used for communication between Control Panel and compute resources

`snmptrap_port*` - port used for snmptrap. This must be greater than 1024

---

We recommend that you do not to change the default value.
In case you change the port value on your OnApp CP - the corresponding change of the port `VM_STATUS_SNMP_PORT` should be made for all Compute resources in `/etc/onapp.conf` file.
allow_incremental_backups - true, if incremental backups are allowed, otherwise false. Incremental backups are not available for Windows virtual servers, as well as under VMware and Solid Fire.

rsync_option_xattrs - true if storing extended attributes is enabled when taking incremental backups.

rsync_option_acls - true if storing access control lists is enabled.

dashboard_api_access_token - the Access token from OnApp Dashboard which is used to synchronize locations between OnApp CP and OnApp Dashboard.

system_theme - the global look and feel theme which is used for the whole cloud by default.

cdn_max_results_per_get_page - the maximum number of results per page delivered when OnApp data are synchronized with Afflexi. The default value is 500.

instance_packages_threshold_num - when the specified number is reached, instance packages are shown in list view in the virtual server creation wizard for easier instance package selection. The default value is 3.

transaction_standby_period - the time which a transaction spends in stand-by period. The default value is 30.

allow_to_collect_errors - 'true' if the Control Panel is allowed to collect, aggregate, encrypt and send crash reports, otherwise 'false'. If this feature is enabled, the error list from your Control Panel will be sent to OnApp in a form of an encrypted API call. By default, this option is disabled.

log_level - log detailization level: debug, info, warn, error and fatal. This parameter is available only for CPs in development mode. It is not displayed for Control Panels in staging or production modes. By default, this parameter is set to 'info'.

graceful_stop_timeout - if the OnApp Engine is stopped, running transactions will fail after the amount of time (seconds) indicated by this parameter. By default, this parameter is set to 300 seconds.

block_size - the block size in MB for disks which is used when migrating disks to another data store. The default value is 8 MB.

draas_enabled - 'true' if DRaaS is enabled for the Cloud; otherwise false

zabbix_host - the IP address of your Zabbix server

zabbix_url - the path to the Zabbix web-interface

zabbix_user - your Zabbix user

zabbix_password - your Zabbix password

enable_hourly_storage_report - generate and send hourly Storage reports. Hourly Storage report is an email with details about storage disks changing degradation statuses (e.g. "Healthy" > "Degraded", "Missing members" > "Healthy", etc). If no such changes occurred during the last hour, the email is not sent.

enable_daily_storage_report - are the health check diagnostic pages sent as email for each Compute zone with storage.

amount_of_service_instances - the number of system processes that perform the OnApp engine tasks simultaneously. Each of the system processes performs the task using a separate CPU core. The default value is 2. Currently, the maximum value is 12. If you input a value larger than 12, the number of system process will still be 12.

migration_rate_limit – the maximum rate limit used for migrating the VS. The default value is 10 Mbps.

recipe_temporary_directory - the temporary recipe directory where all recipe scripts (on Control Panel, compute resources and virtual servers) are generated. The default value is /tmp.
**simultaneous_migrations_per_hypervisor** - the maximum amount of transactions which can be run simultaneously on the target compute resource when migrating a VS. The default value is 5. Applicable only to Migrate VS and Disks.

**snmp_stats_level1_period** - set the delay in seconds between executing the backup tasks which gather information about compute resources uptime and virtual servers’ statuses.

**snmp_stats_level2_period** - set the delay in seconds between executing the backup tasks which gather information about the disk usage, network usage, CPU usage statistics and the list of virtual servers.

**snmp_stats_level3_period** - set the delay in seconds between executing the backup tasks which generate the list of volume groups and logical volumes.

**Page History**

v.6.2 Edge 1
- added the *enable_super_admin_permissions* parameter

v. 6.1 Edge 2
- added the *default_virsh_console_policy* parameter

v. 6.0
- added the following parameters:
  - *default_custom_theme*
  - *drop_firewall_policy_allowed_ips*
  - *default_acceleration_policy*
  - *recipe_temporary_directory*
  - *session_timeout*

v. 5.10
- removed the *enable_huge_pages* parameter

v. 5.9
- added the following parameters:
  - *max_ip_addresses_to_assign_simultaneously*
  - *ip_address_reservation_time*
  - *disable_billing* parameter
  - *transaction_approvals*

v. 5.8
- added the *allow_advanced_vs_management* parameter
v. 5.7
- added the following parameters:
  - support_help_email
  - intra_hypervisor_balance_threshold_ratio
  - inter_hypervisor_balance_threshold_ratio
  - uniform_node_capacity_threshold_ratio
  - pagination_dashboard_pages_limit

v. 5.6
- added the isolated_license parameter
- updated the following parameters:
  - simultaneous_migrations_per_hypervisor
  - migration_rate_limit

v. 5.4
- added the following parameters:
  - block_size
  - migration rate limit
  - simultaneous migrations per hypervisor
  - snmp_stats_level1_period
  - snmp_stats_level2_period
  - snmp_stats_level3_period
- removed ip_range_limit parameter

v. 5.2
- added the following parameters:
  - google_map_token
  - dashboard_stats

- removed the following parameters:
  - system_email
  - system_host
  - system_notification
  - system_support_email
  - email_delivery_method smtp_address
  - smtp_authentication
  - smtp_domain
v. 5.0
- added the following parameters:
  - log_level
  - graceful_stop_timeout

v. 4.2
- added the following parameters:
  - use_yubikey_login
  - yubikey_api_id
  - yubikey_api_key
  - allow_to_collect_errors
  - draas_enabled
  - zabbix_host
  - zabbix_url
  - zabbix_user
  - zabbix_password

v. 4.1
- added the following parameters:
  - instance_packages_threshold_num
  - cdn_max_results_per_get_page
  - transaction_standby_period
  - amount_of_service_instances

v. 3.5
- added system_theme parameter

v. 3.3.1
- added dashboard_api_access_token parameter

v. 3.3
- added instant_stats_period parameter
v. 3.2.2:
- added the following parameters:
  - rsync_option_xattrs
  - rsync_option_acls

v. 3.2:
- added the following parameters:
  - allow_incremental_backups
  - backup_convert_coefficient
  - url_for_custom_tools
- removed wrong_activated_logical_volume_minutes parameter

v. 3.1:
- added the following parameters:
  - allow_start_vms_with_one_ip
  - archive_stats_period
  - is_archive_stats_enabled
  - service_account_name
  - system_alert_reminder_period
  - use_html5_vnc_console
  - wrong_activated_logical_volume_minutes

v. 3.0:
- added the following parameters:
  - enablehuge_pages
  - use_nbd

76.2 Edit System Configuration

To edit the system configuration parameters (System, Backups/Templates, Interface, Defaults) on the /settings/edit page, use the following request:

PUT /settings.xml
PUT /settings.json

XML Request Example

```
curl -i -X PUT http://onapp.test/settings.xml?restart=1 -d '
  <configuration>
    <rsync_option_xattrs>false</rsync_option_xattrs>
    <rsync_option_acls>true</rsync_option_acls>
    <use_yubikey_login>true</use_yubikey_login>
  </configuration>
```
OnApp Cloud 6.3 API Guide

```xml
<gin><yubikey_api_key>00000000000000000000000000000000</yubikey_api_key><yubikey_api_id>0000000</yubikey_api_id><allow_incremental_backups>false</allow_incremental_backups><use_ssh_file_transfer>true</use_ssh_file_transfer><ssh_file_transfer_server>109.123.105.162</ssh_file_transfer_server><ssh_file_transfer_user>root</ssh_file_transfer_user><ssh_file_transfer_options>-oStrictHostKeyChecking=no -oUserKnownHostsFile=/dev/null -oPasswordAuthentication=no</ssh_file_transfer_options><ssh_port>22</ssh_port><template_path>/onapp/templates</template_path><backups_path>/onapp/backups</backups_path><data_path>/onapp/data</data_path><block_size type="integer">40</block_size><update_server_url>http://repo.onapp.com/</update_server_url><generate_comment>Automatically generated by OnApp (3.0.GA)</generate_comment><session_timeout>480</session_timeout><enable_super_admin_permissions>false</enable_super_admin_permissions><simultaneous_backups>2</simultaneous_backups><simultaneous_backups_per_datastore>2</simultaneous_backups_per_datastore><simultaneous_backups_per_hypervisor>1</simultaneous_backups_per_hypervisor><simultaneous_transactions>16</simultaneous_transactions><remote_access_session_start_port>30000</remote_access_session_start_port><remote_access_session_last_port>30099</remote_access_session_last_port><password_enforce_complexity>false</password_enforce_complexity><password_minimum_length>6</password_minimum_length><password_upper_lowercase>false</password_upper_lowercase><password_letters_numbers>false</password_letters_numbers><password_symbols>false</password_symbols><password_force_unique>false</password_force_unique><password_lockout_attempts>3</password_lockout_attempts><password_expiry>1</password_expiry><password_history_length>12</password_history_length><cloud_boot_enabled>false</cloud_boot_enabled><nfs_root_ip>192.168.1.1</nfs_root_ip><cloud_boot_target>192.168.1.1</cloud_boot_target><max_ip_addresses_to_assign_simultaneously>256</max_ip_addresses_to_assign_simultaneously><storage_enabled>false</storage_enabled><cloud_hypervisor_balance_threshold_ratio>95</cloud_hypervisor_balance_threshold_ratio><hypervisor_balance_threshold_ratio>95</hypervisor_balance_threshold_ratio><unifo
```
OnApp Cloud 6.3 API Guide

JSON Request Example

```

```

904
Where:

use_yubikey_login - whether logging in using a Yubikey is enabled

yubikey_api_key - your Yubico secret key

yubikey_api_id - your Yubico client ID

use_ssh_file_transfer - set 1 to allow secure file access, transfer and management to a remote server. It is not possible to utilize SSH file transfer option when incremental backups are enabled.

Skip this option if you are using incremental backups.

If you are using incremental backups option, you should either enable dedicated backup servers in your cloud or share the backups and templates folders (paths) between your compute resources.
ssh_file_transfer_server - the address of the remote server
ssh_file_transfer_user - the login used for remote server authentication. A password is not required, as it is required that you store a host key
ssh_file_transfer_options - SSH protocol options that set the rules and behavior of how to log into the remote server
ssh_port - the port used for contacting SSH servers
template_path - path to the directory where templates will be stored
backups_path - path to the directory where backups will be stored
data_path - path to the directory where the recovery templates will be stored
update_server_url - URL address where OnApp software updates are downloaded from
delete_template_source_after_install - if enabled, the downloaded templates will be deleted after they are distributed
generate_comment - this text is added by OnApp to system configuration files, such as resolv.conf
session_timeout - the timeout between sessions within OnApp in minutes
enable_super_admin_permissions - true, if a super admin feature is enabled; otherwise, false
simultaneous_backups - the maximum allowed number of simultaneous compute resource and data store backup processes
simultaneous_backups_per_datastore - the maximum number of simultaneous data store backup processes
simultaneous_backups_per_hypervisor - the maximum number of simultaneous compute resource backup processes
simultaneous_transactions - the number of transaction runners which the daemon will execute at the same time
simultaneous_backups_per_backup_server - the maximum number of simultaneous backup processes per backup server
guest_wait_time_before_destroy - the VS shutdown period (from 30 to 300 seconds). This allows to refuse the shutdown if the VS is booting and retry every 30 seconds till it is registered.
remote_access_session_start_port - the first port in the range, which are used to remotely connect to virtual servers using the integrated VNC console
remote_access_session_last_port - the last port in the range, which are used to remotely connect to virtual servers using the integrated VNC console
support_help_email - support email to which the help requests will be sent from Control Panel > Help.
ajax_power_update_time - how often VS status is refreshed on the Virtual Servers screen in ms
ajax_pagination_update_time - how often the dashboard, logs and other items are refreshed in ms
hypervisor_live_times - determines how many times the Control Panel server will attempt to contact a compute resource before failover is initiated. 1 compute resource live time = 10 seconds
transaction_approvals - true, if transaction approvals are enabled, otherwise, false
recovery_templates_path - path to the directory where recovery templates will be stored
remove_backups_on_destroy_vm - set 1 to remove all VS backups after this VS was deleted
disable_hypervisor_failover - 1, if compute resource failover will not initiate after meeting the value of the compute_resource_live_times parameter

ips_allowed_for_login - list of IP addresses allowed for login to OnApp CP

Note that Monitis will come to its end of life on June 1st, 2019 and will no longer be supported.

monitis_path - path to the directory where Monitis client (to enable autoscale) will be installed

monitis_account - name of the Monitis account

monitis_apikey - API key to access the Monitis account

locales - an array of locals (the locale code) available for the users

remove_old_root_passwords - 1, if old root password are removed, otherwise 0

pagination_max_items_limit – the maximum number of items after which the Show All option can not be applied.

pagination_dashboard_pages_limit – the maximum number of pages to list log items in the Activity Log section at the main Dashboard page.

allow_advanced_vs_management - true, if the VS advanced configuration is enabled, otherwise, false

default_image_template - default VS template to create a new virtual server

service_account_name - service account name that is automatically created on VMware virtual servers to be able to communicate with them.

default_acceleration_policy - true, if the default acceleration policy is enabled; otherwise, false

default_virsh_console_policy - true, if Virsh console is available by default for all newly created VSs, otherwise, false

default_firewall_policy - default firewall policy for all VSs (unless set otherwise for a particular VS)

drop_firewall_policy_allowed_ips - enter the IP addresses to be allowed as an exception if the default firewall policy is DROP

app_name - application name displayed on the login screen

show_ip_address_selection_for_new_vm - set 1 to enable IP address assignment during VS creation

backup_taker_delay - frequency in seconds for launching the Backup Taker task

billing_stat_updater_delay - frequency in seconds for launching the Billing Stats Monitor task

cluster_monitor_delay - frequency in seconds for launching the Cluster Monitor task

hypervisor_monitor_delay - frequency in seconds for launching the compute resource Monitor task

cdn_sync_delay - frequency in seconds for launching the CDN sync task

google_map_token - add Google API key, if you face the problem with viewing the maps on VS/Smart/Application server creation wizard.

dashboard_stats - an array of statistics, which is shown on the dashboard

schedule_runner_delay - frequency in seconds for launching the Schedule Runner task

transaction_runner_delay - frequency in seconds for launching the Transaction Runner task
zombie_transaction_time – the duration in minutes for the system to consider a task as "pending" before it can be relegated to zombie status

zombie_disk_space_updater_delay – the duration in minutes for the system to check the size of a zombie disk

dns_enabled – 1, if DNS is enabled, otherwise 0

enabled_libvirtAntiSpoofing - 1, if the anti-spoofing protection is enabled, otherwise 0

allow_start_vms_with_one_ip - true, if it is allowed to start up virtual servers with one IP address, otherwise false

disable_billing - true, if billing is disabled, otherwise false

allow_initial_root_password_encryption - set 1 to allow password encryption, otherwise set 0

allow_hypervisor_password_encryption - set 1 to allow VMware compute resource password encryption, otherwise set 0

wipe_out_disk_on_destroy – 1, if the disk wipeout is allowed when deleting or migrating a disk, otherwise 0

password_minimum_length - specify minimum required password length (0-99). The default value is 6.

password_letters_numbers - 1, if the user is enforced to use both letters and numbers in their password, otherwise 0

password_symbols type - 1, if the user is enforced to use symbols in their password, otherwise 0

password_force_unique - 1, if the user is enforced to enter unique password configuration each time they change the password, otherwise 0. This refers to the user account passwords only.

password_lockout_attempts - the number of unsuccessful logon attempts that are allowed before user's account is locked out

password_lockout_response - the message displayed to the user after they exceed the number of unsuccessful logon attempts

password_expiry type - password expiry period in months

password_history_length - the number of last passwords saved in OnApp configuration

force_windows_backups - 1, if forced backup feature is enabled for Windows-based virtual servers, otherwise 0

cloud_boot_enabled - 1, if CloudBoot system is enabled on the cloud, otherwise 0

nfs_root_ip - IP address of the NFS server where templates for creating PXE compute resources are stored

cloud_boot_target - IP of the server where the CP is installed

max_ip_addresses_to_assign_simultaneously - the maximum number of IP addresses that can be assigned to user simultaneously. The default value is 256.

storage_enabled - true, if OnApp storage is enabled, otherwise false

intra_hypervisor_balance_threshold_ratio - percentage ratio to the average free space for all nodes in the hypervisor. The default value is 5%. You will receive a warning in the storage
health check if the free space on one of the compute resource's nodes will be smaller by the indicated value as compared to the other nodes on the compute resource.

`inter_hypervisor_balance_threshold_ratio` - percentage ratio to the average free space for all compute resources within one hypervisor zone. The default value is 5%. You will receive a warning in the storage health check if the free space on one of the compute resources in the zone will be smaller by the indicated value as compared to the other resources in the compute zone.

`uniform_node_capacity_threshold_ratio` - the value (%) by which the size of a node in a data store can differ from the average node size in this data store. The default value is 5%. If this value is exceeded, you will receive a warning in the storage health check.

`prefer_local_reads` - true, if the local read path is enabled, otherwise false

`vmware_cluster_name` - name of the VMware cluster

`ip_address_reservation_time` - the duration in seconds during which the IP address will be reserved for a user and unavailable for other users. The default value is 60.

`isolated_license` - true, if the isolated license is used on the CP, otherwise false

`license_key` - license key of your OnApp CP (required parameter)

`archive_stats_period` - set the time in months for hourly statistics storage. For example, if you set this parameter to 10, the hourly statistics will be stored for the current month and the 10 previous months. And everything older than 10 months will be sent to archive (that is converted into monthly statistics). If this parameter is set as 1, then you can view the detailed hourly statistics for both the current and the previous month.

`instant_stats_period` - set the number of days the instant (raw) statistics will be stored. Starting with OnApp version 5.0, the default value for new installations is 1. For the clouds that have been upgraded from OnApp version 4.3, the default value is 10.

`is_archive_stats_enabled` - set this parameter true to enable archiving of hourly statistics; if false - the statistics will be archived on monthly basis

`use_html5_vnc_console` - true, if the use of HTML 5 console is enabled, otherwise false

---

It is only possible to use HTML 5 console if the Control Panel server is based on CentOS6.

---

`backup_convert_coefficient` - This parameter is for incremental backups only. During the backup conversion to template, backup's size is multiplied by this coefficient to make sure that template will be slightly bigger than the actual size for correct performance.

`snmptrap_addresses`* - a set of IPv4 addresses separated by coma. These IP addresses will be used for communication between Control Panel and compute resources

`snmptrap_port`* - port used for snmptrap. This must be greater than 1024

---

We recommend that you do not to change the default value.
In case you change the port value on your OnApp CP - the corresponding change of the port `VM_STATUS_SNMP_PORT` should be made for all Compute resources in `/etc/onapp.conf` file.

---

`system_theme` - specify the look and feel theme which should be used in the cloud by default. You can set `dark` or `light` parameters.
url_for_custom_tools - path to custom recovery ISO images. Specify the full URL to the tools file packed with GNU Tar + Gzip, like http://domain.com/file.tgz.

allow_incremental_backups - true, if incremental backups are allowed, otherwise false. Incremental backups are not available for Windows virtual servers, as well as under VMware and Solid Fire.

rssync_option_xattrs - true if storing extended attributes is enabled when taking incremental backups.

rssync_option_acls - true if storing access control lists is enabled.

dashboard_api_access_token - the Access token from OnApp Dashboard used to synchronize locations between OnApp CP and OnApp Dashboard.

cdn_max_results_per_get_page - the maximum number of results per page delivered when OnApp data are synchronized with Aflexi. The default value is 500.

instance_packages_threshold_num - when the specified number is reached, instance packages are shown in list view in the virtual server creation wizard for easier instance package selection. The default value is 3.

transaction_standby_period - the time which a transaction spends in stand-by period. The default value is 30.

allow_to_collect_errors - 'true' if the Control Panel is allowed to collect, aggregate, encrypt and send crash reports, otherwise 'false'. If you enable this feature, the error list from your Control Panel will be sent to OnApp in a form of an encrypted API call. By default, this option is disabled.

log_level - log detailization level: debug, info, warn, error and fatal. This parameter is available only for CPs in development mode. It is not displayed for Control Panels in staging or production modes. By default, this parameter is set to 'info'.

graceful_stop_timeout - if the OnApp Engine is stopped, running transactions will fail after the amount of time (seconds) indicated by this parameter. By default, this parameter is set to 300 seconds.

block_size - the block size in MB for disks which is used when migrating disks to another data store. The default value is 8 MB.

draas_enabled - 'true' if DRaaS is enabled for the Cloud; otherwise false

zabbix_host - the IP address of your Zabbix server

zabbix_url - the path to the Zabbix web-interface

zabbix_user - your Zabbix user

zabbix_password - your Zabbix password

amount_of_service_instances - the number of system processes that perform the OnApp engine tasks simultaneously. Each of the system processes performs the task using a separate CPU core. The default value is 2. Currently, the maximum value is 12. If you input a value larger than 12, the number of system process will still be 12.

migration_rate_limit – the maximum rate limit used for migrating the VS. The default value is 10 Mbps.

recipe_temporary_directory - the temporary recipe directory where all recipe scripts (on Control Panel, compute resources and virtual servers) are generated. The default value is /tmp.

simultaneous_migrations_per_hypervisor - the maximum amount of transactions which can be run simultaneously on the target compute resource when migrating a VS. The default value is 5. Applicable only to Migrate VS and Disks.

snmp_stats_level1_period - set the delay in seconds between executing the backup tasks which gather information about compute resources uptime and virtual servers' statuses
OnApp Cloud 6.3 API Guide

snmp_stats_level2_period - set the delay in seconds between executing the backup tasks which gather information about the disk usage, network usage, CPU usage statistics and the list of virtual servers

snmp_stats_level3_period - set the delay in seconds between executing the backup tasks which generate the list of volume groups and logical volumes

Page History
v.6.2 Edge 1
- added the enable_super_admin_permissions parameter

v. 6.1 Edge 2
- added the default_virsh_console_policy parameter

v. 6.0
- added the following parameters:
  - session_timeout
  - drop_firewall_policy_allowed_ips
  - default_custom_theme
  - default_acceleration_policy
  - recipe_temporary_directory

v. 5.10
- removed the enable_huge_pages parameter

v. 5.9
- added the following parameters:
  - max_ip_addresses_to_assign_simultaneously
  - ip_address_reservation_time
  - disable_billing
  - transaction_approvals

v. 5.8
- added the allow_advanced_vs_management parameter

v. 5.7
- added the following parameters:
  - support_help_email
  - intra_hypervisor_balance_threshold_ratio
OnApp Cloud 6.3 API Guide

- `inter_hypervisor_balance_threshold_ratio`
- `uniform_node_capacity_threshold_ratio`
- `pagination_dashboard_pages_limit`

**v. 5.6**
- added the following parameters:
  - `isolated_license`
  - `support_help_email`
- updated the following parameters:
  - `simultaneous.Migrations_per_hypervisor`
  - `migration_rate_limit`

**v. 5.4**
- added the following parameters:
  - `block_size`
  - `migration_rate_limit`
  - `simultaneous.Migrations_per_hypervisor`
  - `snmp_stats_level1_period`
  - `snmp_stats_level2_period`
  - `snmp_stats_level3_period`
- removed `ip_range_limit` parameter

**v. 5.2**
- added the following parameters:
  - `google_map_token`
  - `dashboard_stats`
- removed the following parameters:
  - `system_email`
  - `system_host`
  - `system_notification`
  - `system_support_email`
  - `email_delivery_method.smtp_address`
  - `smtp.authentication`
  - `smtp_domain`
  - `smtp_enable_starttls_auto`
  - `smtp_password`
  - `smtp_port`
  - `smtp_username`
v. 5.0
- added the following parameters:
  - log_level
  - graceful_stop_timeout

v. 4.2
- added the following parameters:
  - use_yubikey_login
  - yubikey_api_id
  - yubikey_api_key
  - allow_to_collect_errors
  - draas_enabled
  - zabbix_host
  - zabbix_url
  - zabbix_user
  - zabbix_password

v. 4.1
- added the following parameters:
  - instance_packages_threshold_num
  - cdn_max_results_per_get_page
  - transaction_standby_period
  - amount_of_service_instances
  - allow_to_collect_errors

v. 3.5
- added system_theme parameter

v. 3.3.1
- added dashboard_api_access_token parameter

v. 3.3
- added the following parameters:
  - delete_template_source_after_install
  - instant_stats_period
  - email_delivery_method
  - smtp_address
  - smtp_authentication
  - smtp_domain
OnApp Cloud 6.3 API Guide

- `smtp_enable_starttls_auto`
- `smtp_password`
- `smtp_port`
- `smtp_username`
- `snmptrap_addresses`
- `snmptrap_port`

- removed `cpu_guarantee` parameter from general system settings; now `cpu_guarantee` can be set for a compute zone

v. 3.2.2

- added the following parameters:
  - `rsync_option_xattrs`
  - `rsync_option_acls`

v. 3.2

- added the following parameters:
  - `allow_incremental_backups`
  - `backup_convert_coefficient`
  - `url_for_custom_tools`

- removed `wrong_activated_logical_volume_minutes` parameter

v. 3.1

- added the following parameters:
  - `allow_start_vms_with_one_ip`
  - `archive_stats_period`
  - `is_archive_stats_enabled`
  - `service_account_name`
  - `system_alert_reminder_period`
  - `use_html5_vnc_console`
  - `wrong_activated_logical_volume_minutes`

v. 3.0

- added the following parameters:
  - `enable_huge_pages`
  - `use_nbd`
77 Service Add-ons

Service Add-ons functionality allows you to offer to your customers additional services on top of your current IaaS Virtual Server offering. This functionality provides the ability to add services to a virtual server.

77.1 Get List of Service Add-ons

To get the list of service add-ons, use the following request:

GET /service_addons.xml
GET /service_addons.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<service_addons type="array">
  <service_addon>
    <id type="integer">2</id>
    <label>service_addon2</label>
    <description nil="true"/>
    <compatible_with type="array">
      <compatible_with>unix</compatible_with>
    </compatible_with>
    <user_id type="integer">2</user_id>
    <icon>
      <url nil="true"/>
    </icon>
    <created_at type="dateTime">2016-12-20T16:32:03+00:00</created_at>
    <updated_at type="dateTime">2016-12-20T16:32:03+00:00</updated_at>
    <available_on_vm_provisioning type="boolean">false</available_on_vm_provisioning>
    <system type="boolean">false</system>
  </service_addon>
<service_addon>...
</service_addons>
```

Where:

- **id** – ID of the service add-on
- **label** – the service add-on title
- **description** – description text added to the service add-on
- **compatible_with** – the OS type, with which this service add-on is compatible
user_id – ID of the user, who created the service add-on
icon – URL with the service add-on icon
created_at – the date when the service add-on was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at — the date when the service add-on was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
available_on_vm_provisioning - true if the ability to choose a service add-on when creating a VS is enabled; otherwise false
system - true, if it is a system service add-on; false, if it is a user service add-on

Page History
v.6.2 Edge 1
• added system parameter
v.5.5
• added available_on_vm_provisioning parameter

77.2 Get Service Add-on Details
To get the details of a service add-on, use the following request:
GET /service_addons/:id.xml
GET /service_addons/:id.json

XML Request Example
```
```

JSON Request Example
```
```

XML Output Example
<service_addon>
  <id type="integer">2</id>
  <label>service_addon2</label>
  <description nil="true"/>
  <compatible_with type="array">
    <compatible_with>unix</compatible_with>
  </compatible_with>
  <user_id type="integer">2</user_id>
  <icon>
    <url nil="true"/>
  </icon>
  <created_at type="dateTime">2016-12-20T16:32:03+00:00</created_at>
  <updated_at type="dateTime">2016-12-20T16:32:03+00:00</updated_at>
  <available_on_vm_provisioning type="boolean">false</available_on_vm_provisioning>
  <system type="boolean">false</system>
</service_addon>

Where:

id – ID of the service add-on

label – the service add-on title

description – description text added to the service add-on

compatible_with – the OS type, with which this service add-on is compatible

user_id – ID of the user, who created the service add-on

icon – URL with the service add-on icon

created_at – the date when the service add-on was created in the
[YYYY][MM][DD][hh][mm][ss]Z format

updated_at — the date when the service add-on was updated in the
[YYYY][MM][DD][hh][mm][ss]Z format

available_on_vm_provisioning - true if the ability to choose a service add-on when creating a
VS is enabled; otherwise false

system - true, if it is a system service add-on; false, if it is a user service add-on

Page History

v.6.2 Edge 1
• added system parameter

v.5.5
• added available_on_vm_provisioning parameter

77.3 Create Service Add-on

To create a service add-on, use the following request:

POST /service_addons.xml
POST /service_addons.json

XML Request Example
OnApp Cloud 6.3 API Guide

```
curl -i -X POST -u user:userpass --url
'Content-type: application/xml' -d
'<service_addon><label>TestServiceAddon</label><description>TestServiceAdd
on</description><available_on_vm_provisioning

type="boolean">false</available_on_vm_provisioning><compatible_with

type="array">unix</compatible_with><system type="boolean">false</system></service_addon>'
```

### JSON Request Example

```
curl -i -X POST -u user:userpass --url
'Content-type: application/json' -d '{"service_addon": {"label":
"TestServiceAddon","available_on_vm_provisioning":false,"description":
"TestServiceAddon", "compatible_with": ["unix"], "system":"false"}}'
```

**Where:**

- **label** – the service add-on title
- **user_id** – ID of the user, who created the service add-on
- **compatible_with** – the OS type, with which this service add-on is compatible (Windows, Unix or both)
- **description** – description text added to the service add-on
- **available_on_vm_provisioning** - true if the ability to choose a service add-on when creating a VS is enabled; otherwise false
- **system** - true, if it is a system service add-on; false, if it is a user service add-on

### Page History

v.6.2
- added the **system** parameter

v.5.5
- added the **available_on_vm_provisioning** parameter

### 77.4 Edit Service Add-on

To edit a service add-on, use the following request:

- **PUT /service_addons/:id.xml**
- **PUT /service_addons/:id.json**

**XML Request Example**
**JSON Request Example**

```
curl -i -X PUT -u user:userpass --url 
'<service_addon><label>TestServiceAddonEDITED</label><description>TestServiceAddonEDITED</description><available_on_vm_provisioning type="boolean">false</available_on_vm_provisioning><compatible_with type="array"><compatible_with>unix</compatible_with><compatible_with>windows</compatible_with></service_addon>'
```

Where:

- **label** – the service add-on title
- **description** – description text added to the service add-on
- **compatible_with** – the OS type, with which this service add-on is compatible (Windows, Unix or both)
- **available_on_vm_provisioning** - true if the ability to choose a service add-on when creating a VS is enabled; otherwise false

**Page History**

v.6.3 Edge 1
- removed the **system** parameter

v.6.2 Edge 1
- added the **system** parameter

v.5.5
- added the **available_on_vm_provisioning** parameter

**77.5 Delete Service Add-on**

To delete a service add-on, use the following request:

DELETE /service_addons/:id.xml
DELETE /service_addons/:id.json

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url 
http://onapp.test/service_addons/2.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**
78 Service Add-on Events

On add events and On remove events menu is a part of the service add-on, where you can configure which actions will be run on the VS, to which the service add-on is assigned, during add-on assignment to the VS and removal.

78.1 Get List of Service Add-on Events

To get the list of service add-on events, use the following request:

GET /service_addons/:service_addon_id/events.xml
GET /service_addons/:service_addon_id/events.json

XML Request Example


JSON Request Example


XML Output Example

<service_addon_events type="array">
  <service_addon_event>
    <id type="integer">4</id>
    <service_addon_id type="integer">5</service_addon_id>
    <recipe_id type="integer">4</recipe_id>
    <action_type>run_recipe</action_type>
    <event_type>on_add_event</event_type>
    <created_at type="dateTime">2016-12-20T16:59:16+00:00</created_at>
    <updated_at type="dateTime">2016-12-20T16:59:16+00:00</updated_at>
  </service_addon_event>
</service_addon_events>

Where:

id – ID of the service add-on event
service_addon_id – ID of the service add-on
recipe_id – ID of the recipe
action_type – the type service add-on event (e.g. run recipe)

event_type – the type of service add-on event (on_add_event, on_remove_event, on_vm_rebuild, on_vm_destroy)

created_at – the date when the service add-on event was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at – the date when the service add-on event was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
78.2 Create Service Add-on Run Recipe Action

To create a service add-on run recipe action, use the following request:

POST /service_addons/:service_addon_id/events/recipes.xml
POST /service_addons/:service_addon_id/events/recipes.json

XML Request Example

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/service_addons/1/events/recipes.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<service_addon_event><destination>run_on_vm</destination><recipe_id>254</recipe_id><event_type>on_add_event</event_type></service_addon_event>
```

JSON Request Example

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/service_addons/1/events/recipes.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"service_addon_event": {"recipe_id": 254, "event_type": "on_add_event", "destination": "run_on_vm"}}
```

Where:

- `recipe_id`* – ID of the recipe
- `event_type`* – the type of service add-on event (on_add_event, on_remove_event, on_vm_rebuild, on_vm_destroy)
- `destination`* - choose the destination where the recipe will be run:
  - set "run_on_vm" if you want to run this recipe action only on VS, to which this service add-on will be assigned
  - set "run_on_cp" if you want to run this recipe action on the whole Control Panel. For more information refer to the Control Panel Recipes Settings

Page History

- v.5.5
  - added destination parameter

78.3 Create Service Add-on Raise Event Action

To create a service add-on raise event action, use the following request:

POST /service_addons/:service_addon_id/events/notifications.xml
POST /service_addons/:service_addon_id/events/notifications.json

XML Request Example

```bash
curl -i -X POST -u user:userpass --url http://onapp.test/service_addons/2/events/notifications.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<service_addon_event><topic_id>31</topic_id><event_type>on_add_event</event_type></service_addon_event>
```
### JSON Request Example

```
curl -i -X POST -u user:userpass --url
http://onapp.test/service_addons/2/events/notifications.json -H 'Accept: application/json' -H 'Content-type: application/json' -d
'{"service_addon_event": {"topic_id": 31, "event_type": "on_add_event"}}'
```

Where:
- **topic_id** – ID of the notification
- **event_type** – the type of service add-on event (on_add_event, on_remove_event, on_vm_rebuild, on_vm_destroy)

### 78.4 Edit Service Add-on Event

To edit a service add-on event, use the following request:

PUT /service_addons/:service_addon_id/events/:id.xml
PUT /service_addons/:service_addon_id/events/:id.json

**XML Request Example**

```
curl -i -X PUT -u user:userpass --url
'<?xml version="1.0" encoding="UTF-8"?>
<service_addon_event><recipe_id>255</recipe_id></service_addon_event>'
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass --url
'{"service_addon_event": {"recipe_id": 255}}'
```

Where:
- **recipe_id** – ID of the recipe

### 78.5 Delete Service Add-on Event

To delete a service add-on event, use the following request:

DELETE /service_addons/:service_addon_id/events/:id.xml
DELETE /service_addons/:service_addon_id/events/:id.json

**XML Request Example**

```
curl -i -X DELETE -u user:userpass --url
```

**JSON Request Example**
curl -i -X DELETE -u user:userpass --url
79 Service Add-on Groups

Service add-on store enables you to organize individual service add-ons into groups that can be used as a paid resource for buckets. This allows you to easily create groups of service add-ons which can be added to the bucket to limit the amount or types of service add-ons that are available to a user.

79.1 Get List of Service Add-on Groups

To get the list of service add-on groups, use the following request:

GET /service_addon_groups.xml
GET /service_addon_groups.json

XML Request Example


JSON Request Example

curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/service_addon_groups.json

XML Output Example
Where:

id – ID of the service add-on group
label – the service add-on group title
parent_id – the ID of the target service add-on group
lft – left nested set identifier
rgt – right nested set identifier
depth – the depth of a given node (distance from this service add-on group to the root)
created_at – the date when the service add-on group was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
updated_at – the date when the service add-on group was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
icon – URL with the service add-on group icon
children – the children pages of service add-on group
relations – the related service add-ons

id – ID of the relation
service_addon_id – ID of the service add-on
service_addon_group_id – ID of the service add-on group
price – the price for service add-on
created_at – the date when the relation was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at – the date when the relation was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
id – ID of the service add-on
title – the service add-on title
description – description text added to the service add-on
compatible_with – the OS type, with which this service add-on is compatible
user_id – ID of the user, who created the service add-on
icon – URL with the service add-on icon
created_at – the date when the service add-on was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at – the date when the service add-on was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
available_on_vm_provisioning - true if the ability to choose a service add-on when creating a VS is enabled; otherwise false

79.2 Get Service Add-on Group Details

To get the details of a particular service add-on group, use the following request:

GET /service_addon_groups/:id.xml
GET /service_addon_groups/:id.json

XML Request Example

curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/service_addon_groups/2.xml

JSON Request Example

curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/service_addon_groups/2.json

XML Output Example
<service_addon_group>
  <id type="integer">3</id>
  <label>demo</label>
  <parent_id nil="true"/>
  <lft type="integer">3</lft>
  <rgt type="integer">4</rgt>
  <depth type="integer">0</depth>
  <created_at type="dateTime">2017-04-20T13:22:55+03:00</created_at>
  <updated_at type="dateTime">2017-04-20T13:22:55+03:00</updated_at>
  <icon>
    <url nil="true"/>
  </icon>
</service_addon_group>

Where:

- **id** – ID of the service add-on group
- **label** – the service add-on group title
- **parent_id** – the ID of the target service add-on group
- **lft** – left nested set identifier
- **rgt** – right nested set identifier
- **depth** – the depth of a given node (distance from this service add-on group to the root)
- **created_at** – the date when the service add-on group was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** – the date when the service add-on group was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **icon** – URL with the service add-on group icon

79.3 Get Service Add-ons Attached to Service Add-on Group

To get the list of service add-ons attached to service add-on group, use the following request:

GET /service_addon_groups/:id/service_addon_group_relations.xml
GET /service_addon_groups/:id/service_addon_group_relations.json

**XML Request Example**

```bash
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/service_addon_groups/2/service_addon_group_relations.xml
```

**JSON Request Example**

```bash
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/service_addon_groups/2/service_addon_group_relations.json
```

**XML Output Example**

```xml
<service_addon_group_relations>
  <service_addon_group id="3" label="demo" parent_id="true" lft="3" rgt="4" depth="0">
    <icon><url/></icon>
  </service_addon_group>
</service_addon_group_relations>
```
<service_addon_group_relations type="array">
    <service_addon_group_relation>
        <id>8</id>
        <service_addon_id>69</service_addon_id>
        <service_addon_group_id>21</service_addon_group_id>
        <price>0.0</price>
        <created_at>2017-06-20T10:55:37.000+03:00</created_at>
        <updated_at>2017-06-20T10:55:37.000+03:00</updated_at>
    </service_addon_group_relation>
</service_addon_group_relations>

Where:

- id – ID of the service add-on group relation
- service_addon_id – ID of the service add-on
- service_addon_group_id – ID of the service add-on group
- price – the price for service add-on
- created_at – the date when the relation was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- updated_at – the date when the relation was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

- id – ID of the service add-on
- label – the service add-on title
- description – description text added to the service add-on
- compatible_with – the OS type, with which this service add-on is compatible
- user_id – ID of the user, who created the service add-on
- icon – URL with the service add-on icon
- created_at – the date when the service add-on was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- updated_at – the date when the service add-on was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

available_on_vm_provisioning - true if the ability to choose a service add-on when creating a VS is enabled; otherwise false

79.4 Create Service Add-on Group

To add a new service add-on group, use the following request:
POST /service_addon_groups.xml
POST /service_addon_groups.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/service_addon_groups.xml -H 'Accept: application/xml' -H
'Content-type: application/xml' -d
'<service_addon_group><label>TestServiceAddonGroup</label><parent_id></parent_id></service_addon_group>'
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/service_addon_groups.json -H 'Accept: application/json' -H
'Content-type: application/json' -d '{"service_addon_group": {"label":
"TestServiceAddonGroup", "parent_id": ""}}'`
```

Where:
- **label** – the service add-on group title
- **parent_id** – ID of the parent group if necessary

### 79.5 Add Service Add-on to Service Add-on Group

To add a service add-on to a service add-on group, use the following request:

POST /service_addon_groups/:id/service_addon_group_relations.xml
POST /service_addon_groups/:id/service_addon_group_relations.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/service_addon_groups/2/service_addon_group_relations.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'<service_addon_group_relation><service_addon_id>69</service_addon_id><price>0</price></service_addon_group_relation>'
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/service_addon_groups/2/service_addon_group_relations.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d
'{"service_addon_group_relation": {"service_addon_id": 69, "price": 0}}'`
```

Where:
- **service_addon_id** – ID of the service add-on
- **price** - the price for service add-on
79.6 Edit Service Add-on Group

To edit a service add-on group, use the following request:

```
PUT /service_addon_groups/:id.xml
PUT /service_addon_groups/:id.json
```

**XML Request Example**

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/service_addon_groups/2.xml -H 'Accept: application/xml'
-H 'Content-type: application/xml' -d
  '<?xml version="1.0" encoding="UTF-8"?>
  <service_addon_group>
    <label>TestServiceAddonGroup</label>
    <parent_id></parent_id>
  </service_addon_group>'
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/service_addon_groups/2.json -H 'Accept: application/json'
-H 'Content-type: application/json' -d
  '{"service_addon_group": {"label": "TestServiceAddonGroup", "parent_id": ""}}'
```

Where:

- `label` – the service add-on group title
- `parent_id` – ID of the parent group if necessary

79.7 Edit Attached Service Add-on

To edit a service add-on attached to a service add-on group, use the following request:

```
PUT /service_addon_groups/:id/service_addon_group_relations/:id.xml
PUT /service_addon_groups/:id/service_addon_group_relations/:id.json
```

**XML Request Example**

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/service_addon_groups/2/service_addon_group_relations/:id.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d
  '<?xml version="1.0" encoding="UTF-8"?>
  <service_addon_group_relation>
    <price>342</price>
  </service_addon_group_relation>'
```

**JSON Request Example**

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/service_addon_groups/2/service_addon_group_relations/:id.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d
  '{"service_addon_group_relation": {"price": 342}}'
```

Where:

- `price` – the price for service add-on
79.8 Delete Service Add-on Group

To delete a service add-on group, use the following request:

DELETE /service_addon_groups/:id.xml
DELETE /service_addon_groups/:id.json

XML Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/service_addon_groups/2.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/service_addon_groups/2.json -H 'Accept: application/json' -H 'Content-type: application/json'
```

79.9 Detach Service Add-on from Service Add-on Group

To detach a service add-on from service add-on group, use the following request:

DELETE /service_addon_groups/:id/service_addon_group_relations/:id.xml
DELETE /service_addon_groups/:id/service_addon_group_relations/:id.json

XML Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/service_addon_groups/2/service_addon_group_relations/23.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request Example

```
curl -i -X DELETE -u user:userpass --url http://onapp.test/service_addon_groups/2/service_addon_group_relations/23.json -H 'Accept: application/json' -H 'Content-type: application/json'
```
80 Templates

A template is a pre-configured operating system image that contains the root directory of an operating system. There are two different kinds of template: system templates and custom templates. System templates are downloaded from the online library. Custom templates are created by backing up an existing virtual server, and converting that backup to a template.

80.1 Get List of All Templates

To get the list of all templates, use the following request:

GET /templates.xml
GET /templates.json

OR

GET /templates/all.xml
GET /templates/all.json

XML Request Example


JSON Request Example


XML Output Example
```xml
<image_templates type="array">
  <image_template>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>f24aece694ffaf125eaf72e9fb13e8dbd</checksum>
    <created_at type="datetime">2012-04-03T14:30:59+00:00</created_at>
    <disk_target_device>--- xen: xvda kvm: hd</disk_target_device>
    <ext4 type="boolean">false</ext4>
    <file_name>centos-6.2-x64-1.0.tar.gz</file_name>
    <id type="integer">1</id>
    <initial_password>Password1</initial_password>
    <initial_username>root</initial_username>
    <label>CentOS 6.2 x64</label>
    <manager_id>centos5.11x64</manager_id>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <operating_system>linux</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>rhel</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <resize_without_reboot_policy/>
    <smart_server type="boolean">true</smart_server>
    <state>inactive</state>
    <template_size type="integer">271308</template_size>
    <updated_at nil="true"/>
    <user_id nil="true"/>
    <version>1.0</version>
    <virtualization type="array">
      <virtualization>xen</virtualization>
      <virtualization>kvm</virtualization>
    </virtualization>
  </image_template>
</image_templates>

Where:

- **allowed_resize_without_reboot** – true if resize without reboot is allowed, otherwise false
- **allowed_hot_migrate** – true if hot migration is allowed, otherwise false
- **allowed_swap** – true if swap is allowed, otherwise false
- **backup_server_id** – the ID of the backup server where the template is stored
- **baremetal_server** - true if the baremetal server can be built from this template
- **cdn** – true if this template can be used for building edge servers. Otherwise false.
- **checksum** – file checksum
- **created_at** – the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **disk_target_device** – the prefix indicating the method of translating the disk to a VS by compute resource
- **ext4** - true if ext4 file system is supported
- **file_name** – the name of the template file
id – ID of template

initial_password - the password set for the VS built on this template

initial_username - the username set for the VS built on this template

label – the template title

manager_id - ID of the template on the template server

min_disk_size – minimum disk size required to build a VS on this template (GB)

min_memory_size – minimum memory size required to build a VS on this template (MB)

operating_system – operating system name

operating_system_arch – architecture of the operating system

operating_system_distro – operating system distribution

operating_system_edition – edition of the OS

operating_system_tail – tail of the OS

parent_template_id – true if this is a system template

remote_id - ID of the template, if it came from the market

resize_without_reboot_policy - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```xml
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
- 4th bit defines the ability to decrease RAM

smart_server - true if a smart server can be built from this template

state – state of the template (active, inactive)

template_size - the size of the template

updated_at — the date when the Network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

user_id - the ID of a user who owns this template

version – version of the file

virtualization – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template.
80.2 Get List of System Templates

To get the list of system templates, use the following request:
GET /templates/system.xml
GET /templates/system.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example
```xml
<image_templates type="array">
  <image_template>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>f24aece694ffaf25eaf72e9fb13e8dbd</checksum>
    <created_at type="datetime">2012-04-03T14:30:59+00:00</created_at>
    <disk_target_device>---xen: xvda kvm: hd</disk_target_device>
    <ext4 type="boolean">false</ext4>
    <file_name>centos-6.2-x64-1.0.tar.gz</file_name>
    <id type="integer">1</id>
    <initial_password>Password1</initial_password>
    <initial_username>root</initial_username>
    <label>CentOS 6.2 x64</label>
    <manager_id>centos5.11x64</manager_id>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <operating_system>linux</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>rhel</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <resize_without_reboot_policy nil="true"/>
    <smart_server type="boolean">true</smart_server>
    <state>inactive</state>
    <template_size type="integer">271308</template_size>
    <updated_at nil="true"/>
    <version>1.0</version>
    <virtualization>xen,kvm</virtualization>
  </image_template>
</image_templates>
```

Where:

- `allowed_resize_without_reboot` – true if resize without reboot is allowed, otherwise false
- `allowed_hot_migrate` – true if hot migration is allowed, otherwise false
- `allowed_swap` – true if swap is allowed, otherwise false
- `backup_server_id` – the ID of the backup server where the template is stored
- `baremetal_server` - true if the baremetal server can be built from this template
- `cdn` – true if this template can be used for building edge servers. Otherwise false.
- `checksum` – file checksum
- `created_at` – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- `disk_target_device` – the prefix indicating the method of translating the disk to a VS by compute resource
- `ext4` - true if ext4 file system is supported
- `file_name` – the name of the template file
- `id` – ID of template
- `initial_password` - the password set for the VS built on this template
**initial_username** - the username set for the VS built on this template

**label** – the template title

**manager_id** - ID of the template on the template server

**min_disk_size** – minimum disk size required to build a VS on this template (GB)

**min_memory_size** – minimum memory size required to build a VS on this template (MB)

**operating_system** – operating system name

**operating_system_arch** – architecture of the operating system

**operating_system_distro** – operating system distribution

**operating_system_edition** – edition of the OS

**operating_system_tail** – tail of the OS

**parent_template_id** – true if this is a system template

**remote_id** - ID of the template, if it came from the market

**resize_without_reboot_policy** - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```xml
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
- 4th bit defines the ability to decrease RAM

**smart_server** - true if a smart server can be built from this template

**state** – state of the template (active, inactive)

**template_size** - the size of the template

**updated_at** — the date when the Network was updated in the [YYYY][MM][DD][T][hh][mm][ss][Z] format

**user_id** - the ID of a user who owns this template

**version** – version of the file

**virtualization** – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

### 80.3 Get List of Own Templates

To get the list of own templates, use the following request:
GET /templates/own.xml
GET /templates/own.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<image_templates type="array">
  <image_template>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>f24aece694ffa125eaf72e9fb13e8dbd</checksum>
    <created_at type="datetime">2012-04-03T14:30:59+00:00</created_at>
    <ext4 type="boolean">false</ext4>
    <file_name>centos-6.2-x64-1.0.tar.gz</file_name>
    <id type="integer">1</id>
    <initial_password>Password1</initial_password>
    <initial_username>root</initial_username>
    <label>CentOS 6.2 x64</label>
    <manager_id>centos5.11x64</manager_id>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <operating_system>linux</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>rhel</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <resize_without_reboot_policy nil="true"/>
    <smart_server type="boolean">true</smart_server>
    <state>inactive</state>
    <template_size type="integer">271308</template_size>
    <updated_at nil="true"/>
    <user_id nil="true"/>
    <version>1.0</version>
    <virtualization>xen,kvm</virtualization>
  </image_template>
</image_templates>
```

Where:

- `allowed_resize_without_reboot` – true if resize without reboot is allowed, otherwise false
allowed_hot_migrate – true if hot migration is allowed, otherwise false
allowed_swap – true if swap is allowed, otherwise false
backup_server_id – the ID of the backup server where the template is stored
baremetal_server - true if the baremetal server can be built from this template
cdn – true if this template can be used for building edge servers. Otherwise false.
checksum – file checksum
created_at – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
disk_target_device – the prefix indicating the method of translating the disk to a VS by compute resource
ext4 - true if ext4 file system is supported
file_name – the name of the template file
id – ID of template
initial_password - the password set for the VS built on this template
initial_username - the username set for the VS built on this template
label – the template title
manager_id - ID of the template on the template server
min_disk_size – minimum disk size required to build a VS on this template (GB)
min_memory_size – minimum memory size required to build a VS on this template (MB)
operating_system – operating system name
operating_system_arch – architecture of the operating system
operating_system_distro – operating system distribution
operating_system_edition – edition of the OS
operating_system_tail – tail of the OS
parent_template_id – true if this is a system template
remote_id - ID of the template, if it came from the market
resize_without_reboot_policy - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```xml
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
4th bit defines the ability to decrease RAM

- `smart_server` - true if a smart server can be built from this template
- `state` – state of the template (active, inactive)
- `template_size` - the size of the template
- `updated_at` — the date when the Network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- `user_id` - the ID of a user who owns this template
- `version` – version of the file
- `virtualization` – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

### 80.4 Get List of User Templates

To get the list of user templates, use the following request:

GET /templates/user.xml
GET /templates/user.json

Contrary to the System templates, the Custom templates parent_template_id parameter indicates the ID of a system template, which has been converted into a custom one.

#### XML Request Example

```bash
```

#### JSON Request Example

```bash
```

#### XML Output Example
<image_templates type="array">
  <image_template>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>f24aece694ffae125eaf72e9fb3e8dbec</checksum>
    <created_at type="datetime">2012-04-03T14:30:59+00:00</created_at>
    <disk_target_device>---:xen: xvda kvm: hd</disk_target_device>
    <ext4 type="boolean">false</ext4>
    <file_name>centos-6.2-x64-1.0.tar.gz</file_name>
    <id type="integer">1</id>
    <initial_password>Password1</initial_password>
    <initial_username>root</initial_username>
    <label>CentOS 6.2 x64</label>
    <manager_id>centos5.11x64</manager_id>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <operating_system linux</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>rhel</operating_system_distro>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <resize_without_reboot_policy></resize_without_reboot_policy>
    <smart_server type="boolean">true</smart_server>
    <state>inactive</state>
    <template_size type="integer">271308</template_size>
    <updated_at nil="true"/>
    <user_id nil="true"/>
    <version>1.0</version>
    <virtualization>xen, kvm</virtualization>
  </image_template>
</image_templates>

Where:

allowed_resize_without_reboot – true if resize without reboot is allowed, otherwise false
allowed_hot_migrate – true if hot migration is allowed, otherwise false
allowed_swap – true if swap is allowed, otherwise false
backup_server_id – the ID of the backup server where the template is stored
baremetal_server - true if the baremetal server can be built from this template
cdn – true if this template can be used for building edge servers. Otherwise false.
checksum – file checksum
created_at – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
disk_target_device – the prefix indicating the method of translating the disk to a VS by compute resource
ext4 - true if ext4 file system is supported
file_name – the name of the template file
id – ID of template
initial_password - the password set for the VS built on this template
**initial_username** - the username set for the VS built on this template

**label** – the template title

**manager_id** - ID of the template on the template server

**min_disk_size** – minimum disk size required to build a VS on this template (GB)

**min_memory_size** – minimum memory size required to build a VS on this template (MB)

**operating_system** – operating system name

**operating_system_arch** – architecture of the operating system

**operating_system_distro** – operating system distribution

**operating_system_edition** – edition of the OS

**operating_system_tail** – tail of the OS

**parent_template_id** – ID of a system template, which has been converted into custom one

**remote_id** - ID of the template, if it came from the market

**resize_without_reboot_policy** - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```xml
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
- 4th bit defines the ability to decrease RAM

**smart_server** - true if a smart server can be built from this template

**state** – state of the template (active, inactive)

**template_size** - the size of the template

**updated_at** — the date when the Network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**user_id** - the ID of a user who owns this template

**version** – version of the file

**virtualization** – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

### 80.5 Get List of Templates of Particular User

To view templates of a particular user, use the following request:
GET /templates/user/:user_id.xml
GET /templates/user/:user_id.json

XML Request Example


JSON Request Example


XML Output Example

<image_templates type="array">
  <image_template>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id nil="true"/>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>f24aece694ffa125eaf72e9fb13e8dbd</checksum>
    <created_at type="datetime">2012-04-03T14:30:59+00:00</created_at>
    <disk_target_device>xen: xvda kvm: hd</disk_target_device>
    <ext4 type="boolean">false</ext4>
    <file_name>centos-6.2-x64-1.0.tar.gz</file_name>
    <id type="integer">1</id>
    <initial_password>Password1</initial_password>
    <initial_username>root</initial_username>
    <label>CentOS 6.2 x64</label>
    <manager_id>centos5.11x64</manager_id>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <operating_system>linux</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>rhel</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <parent_template_id nil="true"/>
    <remote_id nil="true"/>
    <resize_without_reboot_policy nil="true"/>
    <smart_server type="boolean">true</smart_server>
    <state>inactive</state>
    <template_size type="integer">271308</template_size>
    <updated_at nil="true"/>
    <user_id>12</user_id>
    <version>1.0</version>
    <virtualization>xen, kvm</virtualization>
  </image_template>
</image_templates>

Where:

allowed_resize_without_reboot – true if resize without reboot is allowed, otherwise false
allowed_hot_migrate – true if hot migration is allowed, otherwise false
allowed_swap – true if swap is allowed, otherwise false
backup_server_id – the ID of the backup server where the template is stored
baremetal_server - true if the baremetal server can be built from this template
cdn – true if this template can be used for building edge servers. Otherwise false.
checksum – file checksum
created_at – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
disk_target_device – the prefix indicating the method of translating the disk to a VS by compute resource
ext4 - true if ext4 file system is supported
file_name – the name of the template file
id – ID of template
initial_password - the password set for the VS built on this template
initial_username - the username set for the VS built on this template
label – the template title
manager_id - ID of the template on the template server
min_disk_size – minimum disk size required to build a VS on this template (GB)
min_memory_size – minimum memory size required to build a VS on this template (MB)
operating_system – operating system name
operating_system_arch – architecture of the operating system
operating_system_distro – operating system distribution
operating_system_edition – edition of the OS
operating_system_tail – tail of the OS
parent_template_id – ID of a system template, which has been converted into custom one
remote_id - ID of the template, if it came from the market
resize_without_reboot_policy - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```xml
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase CPU
- 2nd bit defines the ability to decrease CPU
- 3rd bit defines the ability to increase RAM
• 4th bit defines the ability to decrease RAM

**smart_server** - true if a smart server can be built from this template

**state** – state of the template (active, inactive)

**template_size** - the size of the template

**updated_at** — the date when the Network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**user_id** - the ID of a user who owns this template

**version** – version of the file

**virtualization** – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

### 80.6 Get List of Inactive Templates

To get the list of inactive templates, use the following request:

GET /templates/inactive.xml
GET /templates/inactive.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
Where:

- **allowed_resize_without_reboot** – true if resize without reboot is allowed, otherwise false
- **allowed_hot_migrate** – true if hot migration is allowed, otherwise false
- **allowed_swap** – true if swap is allowed, otherwise false
- **backup_server_id** – the ID of the backup server where the template is stored
- **baremetal_server** - true if the baremetal server can be built from this template
- **cdn** – true if this template can be used for building edge servers. Otherwise false.
- **checksum** – file checksum
- **created_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **disk_target_device** – the prefix indicating the method of translating the disk to a VS by compute resource
- **ext4** - true if ext4 file system is supported
- **file_name** – the name of the template file
- **id** – ID of template
- **initial_password** - the password set for the VS built on this template
**initial_username** - the username set for the VS built on this template

**label** – the template title

**manager_id** - ID of the template on the template server

**min_disk_size** – minimum disk size required to build a VS on this template (GB)

**min_memory_size** – minimum memory size required to build a VS on this template (MB)

**operating_system** – operating system name

**operating_system_arch** – architecture of the operating system

**operating_system_distro** – operating system distribution

**operating_system_edition** – edition of the OS

**operating_system_tail** – tail of the OS

**parent_template_id** – true if this is a system template

**remote_id** - ID of the template, if it came from the market

**resize_without_reboot_policy** - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```xml
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
- 4th bit defines the ability to decrease RAM

**smart_server** - true if a smart server can be built from this template

**state** – state of the template (active, inactive)

**template_size** - the size of the template

**updated_at** — the date when the Network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**user_id** - the ID of a user who owns this template

**version** – version of the file

**virtualization** – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

### 80.7 Get Template Details

To get the details of a particular template, use the following request:
GET /templates/:template_id.xml
GET /templates/:template_id.json

**XML Request Example**
```
```

**JSON Request Example**
```
```

**XML Output Example**
```
<image_template>
  <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
  <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
  <allowed_swap type="boolean">true</allowed_swap>
  <backup_server_id nil="true"/>
  <baremetal_server type="boolean">true</baremetal_server>
  <cdn type="boolean">false</cdn>
  <checksum>f24aece694ffa125eaf72e9fb13e8dbd</checksum>
  <created_at type="datetime">2012-04-03T14:30:59+00:00</created_at>
  <disk_target_device>--- xen: xvda  kvm: hd</disk_target_device>
  <ext4 type="boolean">false</ext4>
  <file_name>centos-6.2-x64-1.0.tar.gz</file_name>
  <id type="integer">1</id>
  <initial_password>Password1</initial_password>
  <initial_username>root</initial_username>
  <label>CentOS 6.2 x64</label>
  <manager_id>centos5.11x64</manager_id>
  <min_disk_size type="integer">5</min_disk_size>
  <min_memory_size type="integer">128</min_memory_size>
  <operating_system>linux</operating_system>
  <operating_system_arch>x64</operating_system_arch>
  <operating_system_distro>rhel</operating_system_distro>
  <operating_system_tail nil="true"/>
  <parent_template_id nil="true"/>
  <remote_id nil="true"/>
  <resize_without_reboot_policy></resize_without_reboot_policy>
  <smart_server type="boolean">true</smart_server>
  <state>inactive</state>
  <template_size type="integer">271308</template_size>
  <updated_at nil="true"/>
  <user_id nil="true"/>
  <version>1.0</version>
  <virtualization type="array">
    <virtualization>xen</virtualization>
    <virtualization>kvm</virtualization>
  </virtualization>
</image_template>
```

Where:

* **allowed_resize_without_reboot** – true if resize without reboot is allowed, otherwise false
allowed_hot_migrate – true if hot migration is allowed, otherwise false
allowed_swap – true if swap is allowed, otherwise false
backup_server_id – the ID of the backup server where the template is stored
baremetal_server - true if the baremetal server can be built from this template
cdn – true if this template can be used for building edge servers. Otherwise false.
checksum – file checksum
created_at – the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
disk_target_device – the prefix indicating the method of translating the disk to a VS by compute resource
ext4 - true if ext4 file system is supported
file_name – the name of the template file
id – ID of template
initial_password - the password set for the VS built on this template
initial_username - the username set for the VS built on this template
label – the template title
manager_id - ID of the template on the template server
min_disk_size – minimum disk size required to build a VS on this template (GB)
min_memory_size – minimum memory size required to build a VS on this template (MB)
operating_system – operating system name
operating_system_arch – architecture of the operating system
operating_system_distro – operating system distribution
operating_system_edition – edition of the OS
operating_system_tail – tail of the OS
parent_template_id – true if this is a system template
remote_id - ID of the template, if it came from the market
resize_without_reboot_policy - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
• 4th bit defines the ability to decrease RAM

*smart_server* - true if a smart server can be built from this template

*state* – state of the template (active, inactive)

*template_size* – the size of the template

*updated_at* — the date when the Network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

*user_id* - the ID of a user who owns this template

*version* – version of the file

*virtualization* – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template.

### 80.8 Make Template Public

To make a template public, use the following request:

POST /templates/:id/make_public.xml

POST /templates/:id/make_public.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

If a template is queued to be moved to a public list successfully, an HTTP 201 response is returned.

Only Custom templates can be made public.

### 80.9 Delete Template

To delete a template from the system, use the following request:

DELETE /templates/:id.xml

DELETE /templates/:id.json

**XML Request Example**
The system won’t delete the template if it is used by any VSs.

### 80.10 Edit Template

To edit a template, use the following request:

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"image_template":{"label":"zaza_loc12","file_name":"vmwsyfugaqwq6r_20130613175624","version":"1.3","min_disk_size": "10","min_memory_size": "512"}}' --url http://onapp.test/templates/2.json
```

Where you can edit:

- **label** – the template title
- **file_name** – the name of the template file
- **version** – file version
- **min_disk_size** – minimum disk size required to build a VS on this template (GB)
- **min_memory_size** - minimum memory size required to build a VS on this template (MB)
- **allowed_hot_migrate** - true if hot migration for VS, created from this template, is enabled; otherwise false
80.11 Get List of Available for Installation Templates

To get the list of all available for the installation system templates, use the following request:

GET /templates/available.xml
GET /templates/available.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
<remote_templates type="array">
  <remote_template>
    <allow_resize_without_reboot type="boolean">true</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>0996d78608582803b57baae67243664</checksum>
    <disk_target_device>xen: xvda
      kvm: hd</disk_target_device>
    <ext4 type="boolean">true</ext4>
    <file_name>archlinux-2012.08-x64-1.4-xen.kvm.kvm_virtio.tar.gz</file_name>
    <label>Arch Linux 2012.08 x64</label>
    <manager_id>archlinux2012.08x64</manager_id>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">128</min_memory_size>
    <operating_system>linux</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>archlinux</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <resize_without_reboot_policy>
      :xen: 10
      :centos5: 10
      :centos6: 0
      :kvm: 0
      :kvm_virtio: 0
    </resize_without_reboot_policy>
    <smart_server type="boolean">true</smart_server>
    <template_size type="integer">360588383</template_size>
    <version>1.4</version>
    <virtualization>xen, kvm, kvm_virtio</virtualization>
  </remote_template>
</remote_templates>

Where:

**allowed_resize_without_reboot** – true if resize without reboot is allowed, otherwise false

**allowed_hot_migrate** – true if hot migration is allowed, otherwise false

**allowed_swap** – true if swap is allowed, otherwise false

**baremetal_server** - true if the baremetal server can be built from this template

**cdn** – true if this template can be used for building edge servers. Otherwise false.

**checksum** – file checksum

**disk_target_device** – the prefix indicating the method of translating the disk to a VS by compute resource

**ext4** - true if ext4 file system is supported

**file_name** – the name of the template file

**label** – the template title

**manager_id** - ID of the template on the template server

**min_disk_size** – minimum disk size required to build a VS on this template (GB)
min_memory_size – minimum memory size required to build a VS on this template (MB)
operating_system – operating system name
operating_system_arch – architecture of the operating system
operating_system_distro – operating system distribution
operating_system_edition – edition of the OS
operating_system_tail – tail of the OS

resize_without_reboot_policy - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
- 4th bit defines the ability to decrease RAM

smart_server - true if a smart server can be built from this template

template_size - the size of the template

version – version of the file

virtualization – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

### 80.12 Get List of Available for Update Templates

To get the list of all available for the update system templates, use the following request:

GET /templates/upgrades.xml
GET /templates/upgrades.json

**XML Request Example**

```
```

**JSON Request Example**

**XML Output Example**

```xml
<remote_templates type="array">
  <remote_template>
    <allow_resize_without_reboot type="boolean">true</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>978523101ef89278ae59275bdece4b42</checksum>
    <disk_target_device>xen: sda
      kvm: hd</disk_target_device>
    <ext4 type="boolean">true</ext4>
    <file_name>ubuntu-13.04-LAMP-x64-1.2-xen.kvm.kvm_virtio.tar.gz</file_name>
    <label>Ubuntu 13.04 x64 LAMP</label>
    <manager_id>ubuntu13.04lampx64</manager_id>
    <min_disk_size type="integer">256</min_disk_size>
    <min_memory_size type="integer">5</min_memory_size>
    <operating_system>xen,kvm,kvm_virtio</operating_system>
    <version>1.2</version>
    <virtualization>xen,kvm,kvm_virtio</virtualization>
  </remote_template>
  ...</remote_template>
</remote_templates>
```

**Where:**

- `allowed_resize_without_reboot` – true if resize without reboot is allowed, otherwise false
- `allowed_hot_migrate` – true if hot migration is allowed, otherwise false
- `allowed_swap` – true if swap is allowed, otherwise false
- `baremetal_server` - true if the baremetal server can be built from this template
- `cdn` – true if this template can be used for building edge servers. Otherwise false.
- `checksum` – file checksum
- `disk_target_device` – the prefix indicating the method of translating the disk to a VS by compute resource
- `ext4` - true if ext4 file system is supported
- `file_name` – the name of the template file
- `label` – the template title
- `manager_id` - ID of the template on the template server
min_disk_size – minimum disk size required to build a VS on this template (GB)

min_memory_size – minimum memory size required to build a VS on this template (MB)

operating_system – operating system name

operating_system_arch – architecture of the operating system

operating_system_distro – operating system distribution

operating_system_edition – edition of the OS

operating_system_tail – tail of the OS

resize_without_reboot_policy - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:

```
<resize_without_reboot_policy> <xen> <centos5 type="integer">14</centos5>
<centos6 type="integer">14</centos6>
</xen> <kvm> <centos5 type="integer">12</centos5>
<centos6 type="integer">0</centos6>
</kvm>
```

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
- 4th bit defines the ability to decrease RAM

smart_server - true if a smart server can be built from this template

template_size - the size of the template

version – version of the file

virtualization – type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

### 80.13 Install Template

To download and install the available templates, use the following request:

POST http://onapp.test/templates.xml

POST http://onapp.test/templates.json

**XML Request Example**

```
```

**JSON Request Example**

---

957
curl -i -X POST -u 'user:password' http://onapp.test/templates.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"image_template":{"manager_id":"win1010x64profkvm_virtio","backup_server_id":1}}'

Where:
backup_server_id - particular backup server id or use this parameter as empty to install template for all backup servers ("Use ssh file transfer" parameter should be disabled in settings).
manager_id - the remote template attribute

XML Output Example:
Where:

**id** - the id of the template

**label** - the title of the template

**created_at** - the time when the template was created in the [YYYY][MM][DD][hh][mm][ss]Z format
updated_at - the time when the template was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format

version - version of the file

file_name - the name of the template file

operating_system - operating system name

operating_system_distro - operating system distribution

allowed_swap - true if swap is allowed, otherwise false

state - state of the template (active, inactive)

checksum - file checksum

allow_resize_without_reboot - true if resize without reboot is allowed, otherwise false

min_disk_size - minimum disk size required to build a VS on this template (GB)

user_id - the ID of a user who owns this template

template_size - the size of the template

allowed_hot_migrate - true if hot migration for VS, created from this template, is enabled; otherwise false

operating_system_arch - architecture of the operating system

operating_system_edition - edition of the OS

operating_system_tail - tail of the OS

parent_template_id - id of the target template group

virtualization - type of virtualization (xen, kvm or kvm_virtio) which is compatible with this template

min_memory_size - minimum memory size required to build a VS on this template (MB)

disk_target_device - the prefix indicating the method of translating the disk to a VS by compute resource

cdn - true if this template can be used for building edge servers. Otherwise false.

backup_server_id - the id of the backup server

ext4 - true if ext4 file system is supported

smart_server - true if a smart server can be built from this template

baremetal_server - true if a baremetal server can be built from this template

initial_password - the initial password for vCenter-based templates

initial_username - the initial username for vCenter-based templates

remote_id - ID of the template, if it came from the market

manager_id - ID of the template on the template server

resize_without_reboot_policy - all specifically tested templates (all newly added templates and some of the most frequently used ones) will have this parameter which indicates the hot resize possibility for a particular template considering its OS version and virtualization type:
<resize_without_reboot_policy>
  <xen>
    <centos5 type="integer">14</centos5>
    <centos6 type="integer">14</centos6>
  </xen>
  <kvm>
    <centos5 type="integer">12</centos5>
    <centos6 type="integer">0</centos6>
  </kvm>
</resize_without_reboot_policy>

The indicated integer is a decimal representation of a 4-bit binary code, which indicates whether CPU or RAM can (1) or cannot (0) be resized without reboot, where:

- 1st bit defines the ability to increase cpu
- 2nd bit defines the ability to decrease cpu
- 3rd bit defines the ability to increase RAM
- 4th bit defines the ability to decrease RAM

application_server - true if an application server can be built from this template
draas - true if draas is allowed, otherwise false
properties - the attributes of template

### 80.14 Update Template

To update the already installed templates, use the following request:

PUT /templates/:id/upgrade.xml

PUT /templates/:id/upgrade.json

**XML Request Example**

```bash
curl -i -X PUT -u 'user:userpass'
```

**JSON Request Example**

```bash
curl -i -X PUT -u 'user:userpass'
```

**Where:**

*id* - the ID of the required template

### 80.15 Search for Particular Available for Installation Template

To search for a particular available for the installation template, use the following request:

GET /templates/available.xml?search_filter\[query\]=:query

GET /templates/available.json?search_filter\[query\]=:query
XML Request Example

```
curl -i -X GET -u user:userpass
'http://onapp.test/templates/available.xml?search_filter\[query\]=1.5&search_filter\[os\]=FreeBSD&search_filter\[virtualization\]=XEN&search_filter\[arch\]=x64'
```

JSON Request Example

```
curl -i -X GET -u user:userpass
'http://onapp.test/templates/available.json?search_filter\[query\]=bsd&search_filter\[os\]=FreeBSD&search_filter\[virtualization\]=XEN&search_filter\[arch\]=x64'
```

Where:

- **query** - any information from the template label
- **os** - specify the operating system: Linux, Windows, FreeBSD
- **virtualization** - specify the virtualization type: XEN, KVM, Virtio
- **arch** - x64, x86

XML Output Example

```
<remote_templates type="array">
  <remote_template>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <baremetal_server type="boolean">false</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>9446aa5e8d02758191ba62d49f659b0a</checksum>
    <disk_target_device>xen: hd
    kvm: hd</disk_target_device>
    <ext4 type="boolean">false</ext4>
    <file_name>freebsd-9.1-x64-1.5-xen.kvm.kvm_virtio.tar.gz</file_name>
    <manager_id>freebsd9.1x64</manager_id>
    <min_disk_size type="integer">5</min_disk_size>
    <min_memory_size type="integer">256</min_memory_size>
    <operating_system>freebsd</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>freebsd</operating_system_distro>
    <operating_system_edition nil="true"/>
    <smart_server type="boolean">true</smart_server>
    <template_size type="integer">309741781</template_size>
    <url>http://templates.repo.onapp.com/FreeBSD/freebsd-9.1-x64-1.5-xen.kvm.kvm_virtio.tar.gz</url>
    <version>1.5</version>
  </remote_template>
</remote_templates>
```
80.16 Search for Particular Available for Upgrade Template

To search for a particular available for the upgrade template, use the following request:

GET /templates/upgrades.xml?search_filter[query]=:query
GET /templates/upgrades.json?search_filter[query]=:query

XML Request Example

```
curl -i -X GET -u user:userpass
```

JSON Request Example

```
curl -i -X GET -u user:userpass
```

Where:
query - any information from the template label
os - specify the operating system: Linux, Windows, FreeBSD
virtualization - specify the virtualization type: XEN, KVM, Virtio
arch - x64, x86

XML Output Example
<remote_templates type="array">
  <remote_template>
    <allow_resize_without_reboot type="boolean">false</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <baremetal_server type="boolean">true</baremetal_server>
    <cdn type="boolean">false</cdn>
    <checksum>c9230d12ef1391a01e7e39d1ebbf300f</checksum>
    <disk_target_device>xen: xvda
    kvm: hd</disk_target_device>
    <ext4 type="boolean">true</ext4>
    <file_name>cloudlinux-6.4-cPanel-x64-1.3-xen.kvm.kvm_virtio.tar.gz</file_name>
    <label>CloudLinux Server 6.4 x64 cPanel</label>
    <manager_id>cloudlinux6.4cpalex64</manager_id>
    <min_disk_size type="integer">9</min_disk_size>
    <min_memory_size type="integer">512</min_memory_size>
    <operating_system>linux</operating_system>
    <operating_system_arch>x64</operating_system_arch>
    <operating_system_distro>rhel</operating_system_distro>
    <operating_system_edition nil="true"/>
    <operating_system_tail nil="true"/>
    <smart_server type="boolean">true</smart_server>
    <template_size type="integer">2329118640</template_size>
    <url>http://templates.repo.onapp.com/Linux/cloudlinux-6.4-cPanel-x64-1.3-xen.kvm.kvm_virtio.tar.gz</url>
    <version>1.3</version>
    <virtualization>xen, kvm, kvm_virtio</virtualization>
  </remote_template>
</remote_templates>
81 Template Store

Template Store is the class that organizes all VS templates into separate groups. Each template group can be associated with a bucket, in order to control which templates are available to different users.

81.1 Get Template Store Details

To view the list of template groups in the template store, use the following request:

GET http://onapp.test/template_store.xml
GET http://onapp.test/template_store.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<objects type="array">
  <object>
    <created_at type="datetime">2012-05-28T06:02:00-10:00</created_at>
    <depth type="integer">0</depth>
    <id type="integer">1</id>
    <kms type="boolean">false</kms>
    <kms_host type="kms_host"/>
    <kms_port type="kms_port"/>
    <kms_server_label type="kms_server_label"/>
    <label type="label">Linux</label>
    <lft type="integer">1</lft>
    <mak type="boolean">true</mak>
    <own type="boolean">false</own>
    <parent_id nil="true"/>
    <rgt type="integer">28</rgt>
    <updated_at type="datetime">2012-07-13T02:51:56-10:00</updated_at>
  </object>
</objects>
```

Where:

- **objects** - the array with the following parameters:
  - **label** – the group name
  - **created_at** – the date when the template group was created
  - **id** - template group id
**kms_host** – KMS server host name

**kms_port** – KMS server port

**kms_server_label** – KMS server name

**label** - template group name

**mak** - MAK windows licensing type

**own** - user's own license

**parent_id** - id of the target template group

**depth** - the depth of a given node (distance from this template group to the root)

**lft** - left nested set identifier

**rgt** - right nested set identifier

**updated_at** – the date when the template group was updated

### 81.2 Get Template Group Details

To get details of a particular template group, use the following request:

GET /settings/image_template_groups/:image_template_group_id.xml

GET /settings/image_template_groups/:image_template_group_id.json

**XML Request Example:**

```bash
curl -i -X GET -u user:userpass
http://onapp.test/settings/image_template_groups/105.xml
```

**JSON Request Example:**

```bash
curl -i -X GET -u user:userpass
http://onapp.test/settings/image_template_groups/105.json
```

**XML Output Example**

```xml
<image_template_group>
  <created_at type="datetime">2012-07-13T03:25:48-10:00</created_at>
  <depth type="integer">2</depth>
  <id type="integer">105</id>
  <kms type="boolean">false</kms>
  <kms_host></kms_host>
  <kms_port></kms_port>
  <kms_server_label></kms_server_label>
  <lft type="integer">77</lft>
  <mak type="boolean">false</mak>
  <own type="boolean">false</own>
  <parent_id type="integer">102</parent_id>
  <rgt type="integer">78</rgt>
  <updated_at type="datetime">2012-07-13T03:25:48-10:00</updated_at>
</image_template_group>
```

**Where:**
**image_template_group** - the image template array with the following parameters:

- **label** – the group name
- **created_at** – the date when the template group was created
- **id** - template group id
- **kms_host** – KMS server host name
- **kms_port** – KMS server port
- **kms_server_label** – KMS server name
- **label** - template group name
- **mak** - MAK windows licensing type
- **own** - user's own license
- **parent_id** - id of the target template group
- **depth** - the depth of a given node (distance from this template group to the root)
- **lft** - left nested set identifier
- **rgt** - right nested set identifier
- **updated_at** – the date when the template group was updated

### 81.3 Add Template Group

To add a template group, use the following request:

**POST /settings/image_template_groups.xml**

**POST /settings/image_template_groups.json**

**XML Request Example**

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<image_template_group><label>zaza</label><mak>1</mak><kms_host>ededde.fe</kms_host><kms_port>5453</kms_port><kms>1</kms><own>0</own><kms_server_label>wqqwqswq</kms_server_label></image_template_group>' --url http://onapp.test/settings/image_template_groups.xml
```

**JSON Request Example**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"image_template_group":{"label":"zaza","mak":"1","kms_host":"ededde.fe", "kms_port":"5453", "kms":1", "own":1", "kms_server_label":"wqqwqswq"}}' --url http://onapp.test/settings/image_template_groups.json
```

Returns 201 HTTP response on success

**XML Output Example**
Where:

`image_template_group` - the image template array with the following parameters:

- `label` – the group name
- `created_at` – the date when the template group was created
- `id` - template group id
- `kms_host` – KMS server host name
- `kms_port` – KMS server port
- `kms_server_label` – KMS server name
- `label` - template group name
- `mak` - MAK windows licensing type
- `own` - user's own license
- `parent_id` - id of the target template group
- `depth` - the depth of a given node (distance from this template group to the root)
- `lft` - left nested set identifier
- `rgt` - right nested set identifier
- `updated_at` – the date when the template group was updated

### 81.4 Add Child Group

To add a child group to a template group, use the following request:

- POST `http://onapp.test/settings/image_template_groups.xml`
- POST `http://onapp.test/settings/image_template_groups.json`

**XML Request Example**
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '
<?xml version="1.0" encoding="UTF-8"?>
<image_template_group>
  <created_at type="datetime">2012-07-13T03:19:41-10:00</created_at>
  <depth type="integer">1</depth>
  <id type="integer">104</id>
  <kms type="boolean">true</kms>
  <kms_host>ededde.fe</kms_host>
  <kms_port>5453</kms_port>
  <kms_server_label>another</kms_server_label>
  <label>zaza</label>
  <lft type="integer">80</lft>
  <mak type="boolean">true</mak>
  <own type="boolean">false</own>
  <parent_id type="integer">100</parent_id>
  <rgt type="integer">81</rgt>
  <updated_at type="datetime">2012-07-13T03:19:41-10:00</updated_at>
</image_template_group>

Where:
image_template_group - the image template array with the following parameters:

- **label** – the group name
- **created_at** – the date when the group was created
- **id** - template group id
- **kms_host** – KMS server host name
- **kms_port** – KMS server port
- **kms_server_label** – KMS server name
- **mak** - MAK windows licensing type
- **own** - user's own license
- **parent_id** - id of the target template group
- **depth** - the depth of a given node (distance from this group to the root)
- **lft** - left nested set identifier
- **rgt** - right nested set identifier
- **updated_at** – the date when the template group was updated

To edit or delete a child group, use the same requests as for template groups.

### 81.5 Edit Template Group

To edit details of a template group, use the following request:

```
PUT /settings/image_template_groups/:id.xml
PUT /settings/image_template_groups/:id.json
```

**XML Request Example**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
' <image_template_group> <label>zaza</label> <mak>1</mak> <kms_host>ededde.fe</kms_host> <kms_port>5453</kms_port> </image_template_group>
```

http://onapp.test/settings/image_template_groups/100.xml

**JSON Request Example**

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
'{ "image_template_group": {"label": "zaza", "mak": "1", "kms_host": "ededde.fe", "kms_port": "5453", "kms": "1", "own": "1", "kms_server_label": "wqwqw"}}'
```

http://onapp.test/settings/image_template_groups/100.json

Where:

- **label** – the group name
**kms_host** – KMS server host name
**kms_port** – KMS server port
**kms_server_label** – KMS server name
**mak** – MAK windows licensing type
**own** – user’s own license

### 81.6 Delete Template Group

To delete a template group, use the following request:

```
DELETE /settings/image_template_groups/:id.xml
DELETE /settings/image_template_groups/:id.json
```

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/image_template_groups/12.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url
http://onapp.test/settings/image_template_groups/12.xml
```

### 81.7 Get List of Templates Attached to Group

To get the list of templates attached to a template group, use the following request:

```
GET http://onapp.test/settings/image_template_groups/:id/relation_group_templates.xml
GET http://onapp.test/settings/image_template_groups/:id/relation_group_templates.json
```

**XML Request Example**

```bash
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/settings/image_template_groups/98/relation_group_templates.xml
```

**JSON Request Example**

```bash
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/settings/image_template_groups/98/relation_group_templates.json
```

**XML Output Example**
Where:

created_at – the date when the template was created
id – the ID of this relation
image_template_group_id – the ID of template group to which this template is attached
price – the price for the template attached to this template group
template_id – the ID of a template attached to this template group
updated_at – the date when the template group was updated
allow_resize_without_reboot — true if resize without reboot is possible; otherwise false
image_template - an array of image template with the following parameters:
allow_resize_without_reboot - true, if the resize without reboot is allowed, otherwise false
allowed_hot_migrate - true, if the hot migration is allowed, otherwise false
allowed_swap - true, if the swap is allowed, otherwise false
backup_server_id – the ID of the backup server where the template is stored
cdn – true if this template can be used for building edge servers, otherwise false.
checksum – file checksum
disk_target_device – the prefix indicating the method of translating the disk to a VS by compute resource
file_name – the name of the template file
id - the template ID
label - template label
min_disk_size – minimum disk size required to build a VS on this template (GB)
operating_system – operating system name
operating_system_distro – operating system distribution
operating_system_arch – architecture of the operating system
operating_system_edition – edition of the OS
operating_system_tail – tail of the OS
parent_template_id – true if this is a system template
state – state of the template (active, inactive)
template size - the size of the template
updated_at – the date when the template was updated
user_id - the ID of a user who owns this template
version – version of the file
virtualization – type of virtualization (xen or kvm) which is compatible with this template

81.8 Attach Template to Group

To attach a template to a group, use the following request:

POST
/settings/image_template_groups/:image_template_group_id/relation_group_templates.xml
POST
/settings/image_template_groups/:image_template_group_id/relation_group_templates.json

XML Request Example

```
```

JSON Request Example
Returns HTTP 201 response on success.

### 81.9 Detach Template from Group

To detach a template attached to a template group, use the following request:

```
DELETE /settings/image_template_groups/:image_template_group_id/relation_group_templates/:id.xml
DELETE /settings/image_template_groups/:image_template_group_id/relation_group_templates/:id.json
```

**XML Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/settings/image_template_groups/12/relation_group_templates/122.xml
```

**JSON Request Example**

```
curl -i -X DELETE -u user:userpass
http://onapp.test/settings/image_template_groups/12/relation_group_templates/122.json
```
82 Top IOPS Disks

Top IOPS statistics returns 10 disks with top IOPS usage. To view top IOPS statistics, use the Get TOP IOPS Disks request.

82.1 Get TOP IOPS Disks

To view 10 disks with top IOPS usage for the last hour, use the following request:

GET http://onapp.test/top_iops_statistics.xml
GET http://onapp.test/top_iops_statistics.json

XML Request Example

curl -i -X GET -u user:userpass http://onapp.test/top_iops_statistics.xml

JSON Request Example

curl -i -X GET -u user:userpass http://onapp.test/top_iops_statistics.json

XML Output Example

<disk_usage_statistics type="array">
<disk_usage_statistic>
<data_read type="integer">889321</data_read>
<data_written type="integer">90987</data_written>
<disk_id type="integer">1099</disk_id>
</disk_usage_statistic>
...
<disk_usage_statistic>
<data_read type="integer">8</data_read>
<data_written type="integer">0</data_written>
<disk_id type="integer">166</disk_id>
</disk_usage_statistic>
</disk_usage_statistics>

Where:

disk_id - ID of a disk

data_read - number of read I/O operations per second.

data_written - number of written I/O operations per second
83 Transaction Approvals

Transaction approvals functionality lets you set up certain users (approvers) so that they can approve or decline actions performed by other users (requesters). This feature is tied to roles. You can enable the ability to approve transactions for a user role and you can configure that certain transactions performed by a user role will require approval.

83.1 Get Approvals for Role

To get the list of the actions that require approval for a role, use the following request:

GET /roles/:id/approvals.xml
GET /roles/:id/approvals.json

XML Request Example

```bash
curl -i -X GET -u user:userpass --url
http://onapp.test/roles/156/approvals.xml
-H 'Accept: application/xml' -H
'Content-type: application/xml'
```

JSON Request Example

```bash
curl -i -X GET -u user:userpass --url
http://onapp.test/roles/156/approvals.json
-H 'Accept: application/json' -H
'Content-type: application/json'
```

XML Output Example

```xml
<hash>
  <transaction_action_approvals type="array">
    <transaction_action_approval>
      <id type="integer">1</id>
      <action>destroy_disk</action>
      <enabled type="Boolean">false</enabled>
    </transaction_action_approval>
    ...<transaction_action_approval>
  </transaction_action_approvals>
</hash>
```

Where:

- **id** - ID of the approval
- **action** - the name of the action that requires/does not require approval
- **enabled** - ‘true’ if this action requires approval if performed by this user role, otherwise ‘false’

83.2 Get List of Transactions Pending Approval

To get the list of the transactions that require approval, use the following request:

GET /approvals.xml
GET /approvals.json
XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<approvals type="array">
  <approval>
    <id type="integer">511</id>
    <user_id type="integer">53</user_id>
    <status>approved</status>
    <created_at type="dateTime">2017-07-14T16:40:54+03:00</created_at>
    <updated_at type="dateTime">2017-07-14T16:40:54+03:00</updated_at>
    <log_item_id type="integer">69712</log_item_id>
  </approval>
  ...
</approvals>
```

Where:

- **id** – transaction ID
- **user_id** - the ID of the user who has requested this transaction
- **status** - the status of the transaction: approved, declined or pending
- **created_at** - the time in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** - the time in the [YYYY][MM][DD][hh][mm][ss]Z format
- **log_item_id** - the ID of the log related to the transaction

### 83.3 Set Approvals for Role

To set the list of the transactions that will require approval for a user role, use the following request:

- **PUT /roles/:id/approvals.xml**
- **PUT /roles/:id/approvals.json**

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass --url http://onapp.test/roles/13/approvals.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<transaction_action_approval_ids type="array"><value type="integer">1</value><value type="integer">2</value></transaction_action_approval_ids>'
```

**JSON Request Example**
curl -i -X PUT -u user:userpass --url http://onapp.test/roles/13/approvals.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"transaction_action_approval_ids": [1, 2]}'

Where:
The first number indicates the ID of the action for which approval is set. The second number indicates whether approval is enabled for the action ("1") or not ("0"). For the actions which you do not mention in the request, approval is disabled by default.
To learn the ID of the action for which approval can be set, use the following request:
GET /roles/:id/set_approvals.xml
GET /roles/:id/set_approvals.json
For more information on this request, refer to Get Approvals for Role.

83.4 Approve Transaction

To approve a transaction, use the following request:
PUT /logs/:id/approve.xml
PUT /logs/:id/approve.json
XML Request Example


JSON Request Example


Specify the ID of the log related to the transaction which you wish to approve.

83.5 Decline Transaction

To decline a transaction, use the following request:
PUT /logs/:id/decline.xml
PUT /logs/:id/decline.json
XML Request Example

JSON Request Example

```bash
```

Specify the ID of the log related to the transaction which you wish to decline.
84 Transactions

This class represents all the operations happening in your cloud, such as VS provisioning, OS configuring, VS start up, operations with disks, and so on.

Currently the following behavior is implemented in OnApp:

- **GET /transactions.xml** and **GET /transactions.json** requests return the last 10 transactions without "log_output" parameter.
- **GET /transactions.xml/page/2** and **GET /transactions.json/page/2** requests return the next 10 transactions.
- **Use the GET /transactions.xml/per_page/20** and **GET /transactions.json/per_page/20** requests to change count of returned transaction (in other words - use pagination).
- **Use the GET /transactions.xml?detailed** and **GET /transactions.json?detailed** requests to see "log_output" details (pagination also works here).

84.1 Get List of Transactions

To view the list of transactions without log output, use the following request:

GET /transactions.xml
GET /transactions.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example:**
<transactions type="array">
 transaction
     <action>stop_virtual_machine</action>
     <actor nil="true"/>
     <allowed_cancel type="boolean">true</allowed_cancel>
     <associated_object_id type="integer">7390</associated_object_id>
     <associated_object_type>VirtualMachine</associated_object_type>
     <created_at type="datetime">2015-03-19T17:33:58+03:00</created_at>
     <dependent_transaction_id nil="true"/>
     <id type="integer">101666</id>
     <identifier>06f13xq65pk699</identifier>
     <params>
         <shutdown_type type="symbol">soft</shutdown_type>
     </params>
     <parent_id type="integer">7390</parent_id>
     <parent_type>VirtualMachine</parent_type>
     <pid type="integer">9671</pid>
     <priority type="integer">10</priority>
     <start_after type="datetime">2015-03-19T17:33:58+03:00</start_after>
     <started_at type="datetime">2015-03-19T17:33:59+03:00</started_at>
     <status>complete</status>
     <updated_at type="datetime">2015-03-19T17:34:10+03:00</updated_at>
     <user_id type="integer">45</user_id>
 </transaction>
 ...
 transaction
     <action>destroy_user</action>
     <actor nil="true"/>
     <allowed_cancel type="boolean">true</allowed_cancel>
     <associated_object_id nil="true"/>
     <associated_object_type nil="true"/>
     <created_at type="datetime">2015-03-19T15:12:19+03:00</created_at>
     <dependent_transaction_id nil="true"/>
     <id type="Integer">101607</id>
     <identifier>kae7h1vxt2sp89</identifier>
     <params>
         <with_destroy type="boolean">false</with_destroy>
     </params>
     <parent_id type="integer">75</parent_id>
     <parent_type>User</parent_type>
     <pid type="integer">9671</pid>
     <priority type="integer">10</priority>
     <start_after type="datetime">2015-03-19T15:12:19+03:00</start_after>
     <started_at type="datetime">2015-03-19T15:12:20+03:00</started_at>
     <status>complete</status>
     <updated_at type="datetime">2015-03-19T15:16:33+03:00</updated_at>
     <user_id nil="true"/>
 </transaction>
</transactions>

Where:

pid — external process ID

created_at — the time when the record of transaction was made in the database, in the [YYYY][MM][DD][T][hh][mm][ss]Z format

start_after — the time after which the transaction may start, in the [YYYY][MM][DD][T][hh][mm][ss]Z format

finished_at — reserved detail

updated_at — the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

actor — reserved detail

priority — priority of the transaction (reserved detail)
**parent_type** — the type of the transaction target (virtual server, disk or compute resource)

**action** — the type of transaction performed

**id** — transaction ID

**user_id** — ID of the user who performed the transaction

**dependent_transaction_id** — ID of the transaction that the current transaction depends on. For independent transactions this remains empty.

**allowed_cancel** — true if cancellation is allowed. Otherwise false.

**parent_id** — ID of the target VS, disk or compute resource

**started_at** — time when the transaction was started, in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**params** — parameters of the transaction

**log_output** — an array with log output details

**status** — status of the transaction (complete, failed, pending, etc)

**identifier** — identifier of the virtual server

### 84.2 Get List of VS Transactions without Log Output

To get the list of VS transactions without log output, use the following request:

GET /virtual_machines/:virtual_machine_id/transactions.xml
GET /virtual_machines/:virtual_machine_id/transactions.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**
Where:

**pid** — external process ID

**created_at** — the time when the record of transaction was made in the database, in the `[YYYY][MM][DD][hh][mm][ss]Z` format

**start_after** — the time after which the transaction may start, in the `[YYYY][MM][DD][hh][mm][ss]Z` format

**finished_at** — reserved detail

**updated_at** — the date in the `[YYYY][MM][DD][hh][mm][ss]Z` format

**actor** — reserved detail

**priority** — priority of the transaction (reserved detail)

**parent_type** — type of the transaction target (virtual server, disk or compute resource)

**action** — the type of transaction performed

**id** — transaction ID

**user_id** — ID of the user who performed the transaction

**dependent_transaction_id** — ID of the transaction that the current transaction depends on. For independent transactions this remains empty.

**allowed_cancel** — true if cancellation is allowed. Otherwise false.

**parent_id** — ID of the target virtual server, disk or compute resource

**started_at** — time when the transaction was started in the `[YYYY][MM][DD][hh][mm][ss]Z` format

**params** — parameters of the transaction

**log_output** — an array with log output details

**status** — status of the transaction (complete, failed, pending, etc)
**identifier** — identifier of the virtual server

### 84.3 Get List of Transactions with Log Output

To view the list of transactions with log output, use the following request:

GET /transactions.xml?detailed
GET /transactions.json?detailed

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example:**

---

OnApp Cloud 6.3 API Guide
<transactions type="array">
  <transaction>
    <action>destroy_virtual_machine</action>
    <actor nil="true"/>
    <allow_cancel type="boolean">true</allow_cancel>
    <associated_object_id type="integer">7390</associated_object_id>
    <associated_object_type>VirtualMachine</associated_object_type>
    <created_at type="datetime">2015-03-19T18:04:44+03:00</created_at>
    <dependent_transaction_id nil="true"/>
    <id type="integer">101673</id>
    <identifier>o6fl3xq65pk699</identifier>
    <log_output>
      # Logfile created on 2015-03-19 11:04:47 -0400 by logger.rb /41954
      Remote Server: 192.168.7.41
      Running: Storage API Call: PUT 192.168.7.41:8080/lvm/Datastore/onapp-qm20\inver40vmz/VDisk/pmh0q4dm6o2ypt "{\"state\":3}"
      {"result"="SUCCESS"}
      Remote Server: 192.168.7.41
      ... Running: rm -f /onapp/config/o6f13xq65pk699*
      Running: rm -f /onapp/firewall-rules/odamguc5ndbp44*
      Running: rm -f /onapp/firewall-rules/ip6-odamguc5ndbp44*
    </log_output>
    <params>
      <remote_ip>194.44.160.178</remote_ip>
      <destroy_msg>Destroy from Web interface</destroy_msg>
      <skip_notification type="boolean">true</skip_notification>
    </params>
  </transaction>
  <transaction>
    <parent_id type="integer">7390</parent_id>
    <parent_type>VirtualMachine</parent_type>
    <pid type="integer">9671</pid>
    <priority type="integer">10</priority>
    <start_after type="datetime">2015-03-19T18:04:44+03:00</start_after>
    <started_at type="datetime">2015-03-19T18:04:47+03:00</started_at>
    <status>complete</status>
    <updated_at type="datetime">2015-03-19T18:05:00+03:00</updated_at>
    <user_id type="integer">45</user_id>
  </transaction>
  ...
  ...
</transactions>

Where:

- **pid** — external process ID
- **created_at** — the time when the record of transaction was made in the database, in the [YYYY][MM][DD][hh][mm][ss]Z format
- **start_after** — the time after which the transaction may start, in the [YYYY][MM][DD][hh][mm][ss]Z format
- **finished_at** — reserved detail
- **updated_at** — the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **actor** — reserved detail
- **priority** — priority of the transaction (reserved detail)
- **parent_type** — the type of the transaction target (virtual server, disk or compute resource)
**action** — the type of transaction performed

**id** — transaction ID

**user_id** — ID of the user who performed the transaction

**dependent_transaction_id** — ID of the transaction that the current transaction depends on. For independent transactions this remains empty.

**allowed_cancel** — true if cancellation is allowed. Otherwise false.

**parent_id** — ID of the target VS, disk or compute resource

**started_at** — time when the transaction was started, in the [YYYY][MM][DD][hh][mm][ss]Z format

**params** — parameters of the transaction

**log_output** — an array with log output details

**status** — status of the transaction (complete, failed, pending, etc)

**identifier** — identifier of the virtual server

### 84.4 Get Transaction Details

To view a transaction's details with log output, use the following request:

GET /transactions/:id.json
GET /transactions/:id.xml

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
<transaction>
  <action>create_edge_server</action>
  <actor nil="true"/>
  <allowed_cancel type="boolean">true</allowed_cancel>
  <associated_object_id type="integer">7427</associated_object_id>
  <associated_object_type>VirtualMachine</associated_object_type>
  <created_at type="datetime">2015-03-19T05:42+03:00</created_at>
  <dependent_transaction_id type="integer">101682</dependent_transaction_id>
  <id type="integer">101683</id>
  <identifier>ipj39mncfmlc8l</identifier>
  <log_output>
    # Logfile created on 2015-03-19 11:06:30 -0400 by logger.rb/41954
    Remote Server: 109.123.91.23
  </log_output>
  <params>
    <add_to_marketplace type="boolean">true</add_to_marketplace>
    <skip_notification type="boolean">true</skip_notification>
  </params>
  <parent_id type="integer">7427</parent_id>
  <parent_type>VirtualMachine</parent_type>
  <pid type="integer">9671</pid>
  <priority type="integer">10</priority>
  <start_after type="datetime">2015-03-19T05:42+03:00</start_after>
  <started_at type="datetime">2015-03-19T06:30+03:00</started_at>
  <status>running</status>
  <updated_at type="datetime">2015-03-19T06:30+03:00</updated_at>
  <user_id type="integer">45</user_id>
</transaction>

For details, refer to the Get List of Transactions section.

### 84.5 Get Transaction Details Without Log Output

To view transaction's details without log output, use the following request:

**GET /transactions/:id.json?short**

**GET /transactions/:id.xml?short**

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
For details refer to the [Get List of Transactions](#) section.
85 User Additional Fields

User Additional Fields allow administrators to create custom fields and use them to create/edit additional information in a user's profile.

85.1 Get List of Additional Fields

To get the list of user additional fields, use the following request:

GET /user_additional_fields.xml
GET /user_additional_fields.json

XML Request Example

```
curl -i -X GET -u user:userpass
http://onapp_test/user_additional_fields.xml
```

JSON Request Example

```
curl -i -X GET -u user:userpass
http://onapp_test/user_additional_fields.json
```

XML Output Example

```
<user_additional_fields type="array">
  <user_additional_field>
    <name>test_edited_by_api</name>
    <default_value>0</default_value>
    <data_type>integer</data_type>
    <id type="integer">1</id>
  </user_additional_field>
</user_additional_fields>
```

Where:

- **name** – the additional field name
- **default_value** – information which will be displayed if the user hasn't specified information for this field, or if they enter information that doesn't match the preset data type
- **data_type** - integer/string additional field data type
- **id** – the additional field id

85.2 Get Additional Field Details

To get details of a particular additional field, use the following request:

GET /user_additional_fields/:id.xml
GET /user_additional_fields/:id.json

XML Request Example
85.3 Add Additional Field

To create a user additional field, use the following request:

**POST** /user_additional_fields.xml
**POST** /user_additional_fields.json

**XML Request Example**

```
curl -i -X POST -d
'\n<user_additional_field><name>TestField</name><data_type>string</data_type>
<default_value>testvalue</default_value><id>1</id></user_additional_field>'
-u user:userpass http://onapp_test/user_additional_fields.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X POST -d
'\n{"user_additional_field":{"name":"TestField","data_type":"string","default_value":"TestName"}}' -u user:userpass
http://onapp_test/user_additional_fields.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

*name* - the unique Additional Field name. (maximum length is 256 characters)

*data_type* - integer/string additional field data type
**default_value** - information which will be displayed if the user hasn't specified information for this field, or if they enter information that doesn't match the preset data type. (maximum length 256 symbols.)

### 85.4 Edit Additional Field

To edit a user additional field, use the following request:

PUT /user_additional_fields.xml
PUT /user_additional_fields.json

**XML Request Example**

```bash
curl -i -X PUT -d
  '<user_additional_field><name>TestField</name><data_type>string</data_type>
  <default_value>CHANGEDvalue</default_value></user_additional_field>'
  -u user:userpass http://onapp_test/user_additional_fields/:field_id.xml
  -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X PUT -d
  '{"user_additional_field":{"name":"TestField","data_type":"string","default_value":"TestName"}}'
  -u user:userpass http://onapp_test/user_additional_fields/:field_id.json
  -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

- **name** - the unique Additional Field name. (maximum length is 256 characters)
- **data_type** - integer/string additional field data type
- **default_value** - information which will be displayed if the user hasn't specified information for this field, or if they enter information that doesn't match the preset data type. (maximum length 256 symbols.)

### 85.5 Delete Additional Field

To delete a user additional field, use the following request:

DELETE /user_additional_fields.xml
DELETE /user_additional_fields.json

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass
  http://onapp_test/user_additional_fields/:id.xml
  -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**
85.6 Search User by Additional Field Parameter

To find a user who is assigned a particular additional field, use the following request:

GET /users/field_name=field_value.xml
GET /users/field_name=field_value.json

**XML Request Example:**

```bash
curl -i -X GET -u user:userpass
http://onapp.test/users/field_name=field_value.xml
```

**JSON Request Example:**

```bash
curl -i -X GET -u user:userpass
http://onapp.test/users/field_name=field_value.json
```

Where:

- `field_name` – the name of the additional field which is assigned to the user in search
- `field_value` – the value set for the specified additional field for this particular user

The result of the search request will be the list of users with their details who are assigned the additional field `field_name` with the values starting with `field_value`. 
86 User Groups

User groups enable you to associate users into groups. So far user groups are used to apply a particular theme to a group of users.

86.1 Get List of User Groups

To get the list of user groups, use the following request:

GET /user_groups.xml
GET /user_groups.json

**XML Request Example**

```
curl -i -u user:userpass -X GET http://onapp.test/user_groups.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -u user:userpass -X GET http://onapp.test/user_groups.json -H 'Accept: application/json' -H 'Content-type: application/json'
```

**XML Output Example**
<user_groups type="array">
  <user_group>
    <closed type="boolean">false</closed>
    <created_at type="datetime">2015-04-15T16:02:20+03:00</created_at>
    <federation_enabled type="boolean">false</federation_enabled>
    <federation_id nil="true"/>
    <hypervisor_id nil="true"/>
    <identifier>abywlogotbqza</identifier>
    <label>test</label>
    <traded type="boolean">false</traded>
    <updated_at type="datetime">2015-04-15T16:02:20+03:00</updated_at>
    <bucket_id nil="true"/>
    <user_buckets type="array">
      <user_bucket>
        <id type="integer">1</id>
        <label>default billing</label>
        <created_at type="datetime">2013-09-03T15:31:30+03:00</created_at>
        <updated_at type="datetime">2013-09-03T15:31:30+03:00</updated_at>
        <currency_code>USD</currency_code>
        <show_price nil="true"/>
        <monthly_price type="decimal">0.0</monthly_price>
        <allows_mak type="boolean">true</allows_mak>
        <allows_kms type="boolean">false</allows_kms>
        <allows_own type="boolean">false</allows_own>
      </user_bucket>
    </user_buckets>
    <roles type="array">
      <role>
        <id type="integer">1</id>
        <label>Administrator</label>
        <identifier>admin</identifier>
        <created_at type="datetime">2013-09-03T15:31:13+03:00</created_at>
        <updated_at type="datetime">2015-04-15T15:26:46+03:00</updated_at>
        <users_count type="integer">24</users_count>
      </role>
    </roles>
  </user_group>
</user_groups>

Where:

closed - not relevant to user groups
created_at – the date when this record was created in database
federation_enabled - not relevant to user groups
federation_id - not relevant to user groups
id – the group ID
identifier —identifier of the user group
label – the group name
traded - not relevant to user groups
updated_at – the date when this record was updated in database
bucket_id - the ID of the bucket which has been assigned to this user group
user_buckets - an array of buckets to which the users in this group are assigned, where
  id - the billing type ID
  label - the bucket name
**created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

**updated_at** - the date when the bucket was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

**currency_code** - the currency in which the users are charged

**show_price** - true, if users can see the prices set up for them, otherwise false

**monthly_price** - monthly fee for plan usage

**allows_kms** - true, if the KMS licensing is allowed for this bucket, otherwise false

**allows_mak** - true, if the MAK licensing is allowed, otherwise false

**allows_own** - true, if adding own licenses is allowed for this bucket, otherwise false

**roles** — an array of user roles to which this account is assigned to, where:

**created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

**id** – role ID

**label** – role title

**identifier** – role identifier

**created_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format

**updated_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

**users_count** - the number of users assigned to the role

### 86.2 Get User Group Details

To get details for a particular user group, use the following request:

GET /user_groups/:id.xml

GET /user_groups/:id.json

**XML Request Example**

```
curl --user user:password -X GET http://onapp.test/user_groups/56.xml -H 'Accept: application/xml' -H 'Content-Type: application/xml'
```

**JSON Request Example**

```
curl --user user:password -X GET http://onapp.test/user_groups/56.json -H 'Accept: application/json' -H 'Content-Type: application/json'
```

**XML Output Example**

---
<user_group>
<closed type="boolean">false</closed>
<created_at type="datetime">2015-04-15T16:02:20+03:00</created_at>
<federation_enabled type="boolean">false</federation_enabled>
<federation_id nil="true"/>
<hypervisor_id nil="true"/>
<id type="integer">56</id>
<identifier>abywlogotbqza</identifier>
<label>test</label>
<traded type="boolean">false</traded>
<updated_at type="datetime">2015-04-15T16:02:20+03:00</updated_at>
<bucket_id nil="true"/>
<user_buckets type="array">
  <user_bucket>
    <id type="integer">1</id>
    <label>default billing</label>
    <created_at type="datetime">2013-09-03T15:31:30+03:00</created_at>
    <updated_at type="datetime">2013-09-03T15:31:30+03:00</updated_at>
    <currency_code>USD</currency_code>
    <show_price nil="true"/>
    <discount_due_to_free>0.0</discount_due_to_free>
    <total_amount_with_discount>0.0</total_amount_with_discount>
    <monthly_price type="decimal">0.0</monthly_price>
    <allows_mak type="boolean">true</allows_mak>
    <allows_kms type="boolean">false</allows_kms>
    <allows_own type="boolean">false</allows_own>
  </user_bucket>
</user_buckets>
<roles type="array">
  <role>
    <id type="integer">1</id>
    <label>Administrator</label>
    <identifier>admin</identifier>
    <created_at type="datetime">2013-09-03T15:31:13+03:00</created_at>
    <updated_at type="datetime">2015-04-15T15:26:46+03:00</updated_at>
    <users_count type="integer">24</users_count>
  </role>
</roles>
</user_group>

Where:

- **closed**: not relevant to user groups
- **created_at**: the date when this record was created in database
- **federation_enabled**: whether Federation is enabled for the compute zone
- **id**: the group ID
- **identifier**: identifier of the user group
- **label**: the group name
- **traded**: not relevant to user groups
- **updated_at**: the date when this record was updated in database
- **bucket_id**: the ID of the bucket which has been assigned to this user group
- **user_buckets**: an array of buckets to which the users in this group are assigned, where
  - **id**: the billing type ID
  - **label**: the bucket name
  - **created_at**: the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
updated_at - the date when the bucket was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

currency_code - the currency in which the users are charged

show_price - true, if users can see the prices set up for them, otherwise false

total_amount_with_discount - the price of used resources that excludes the cost of the resources that were created within the bucket's free limits.

discount_due_to_free - the price of the resources that were created within the buckets' free limits.

monthly_price - monthly fee for plan usage

allows_kms - true, if the KMS licensing is allowed for this bucket, otherwise false

allows_mak - true, if the MAK licensing is allowed, otherwise false

allows_own - true, if adding own licenses is allowed for this bucket, otherwise false

roles — an array of user roles to which this account is assigned to, where

created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

id – role ID

label – role title

identifier – role identifier

created_at – the date in the [YYYY][MM][DD][hh][mm][ss]Z format

updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

users_count - the number of users assigned to the role

86.3 Get List of Users Assigned to User Group

To get details for a particular user group, use the following request:

GET /user_groups/:id/users.xml
GET /user_groups/:id/users.json

XML Request Example

curl -i -u user:userpass -X GET http://onapp.test/user_groups/45/users.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -u user:userpass -X GET
http://onapp.test/user_groups/45/users.json -H 'Accept: application/json'
-H 'Content-type: application/json'

XML Output Example
<users type="array">

<user>

<activated_at type="datetime">2012-04-03T14:30:34+00:00</activated_at>
<avatar nil="true"/>
<billing_plan_id type="integer">201</billing_plan_id>
<cdn_account_status>ACTIVE</cdn_account_status>
<cdn_status>ACTIVE</cdn_status>
<created_at type="datetime">2012-04-03T14:30:34+00:00</created_at>
<deleted_at nil="true"/>
<email>admin@example.com</email>
<firewall_id type="integer">4</firewall_id>
<first_name>John</first_name>
<group_id nil="true"/>
<id type="integer">1</id>
<image_template_group_id nil="true"/>
<infoboxes>
<hidden_infoboxes type="array">
<hidden_infobox>4840313084eef1f8e1dada293eb1b1ae</hidden_infobox>
</hidden_infoboxes>
</infoboxes>
<last_name>Smith</last_name>
<locale>en</locale>
<login>admin</login>
<password_changed_at type="datetime">2014-08-21T16:00:15+03:00</password_changed_at>
<registered_yubikey>true</registered_yubikey>
<status>active</status>
<supplied type="boolean">false</supplied>
<suspend_at nil="true"/>
<payment_amount type="decimal">0.0</payment_amount>
<roles type="array">

<role>

<created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
<id type="integer">1</id>
<identifier>admin</identifier>
<label>Administrator</label>
<permissions type="array">

<permission>

<created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
<id type="integer">110</id>
<identifier>autobackup_templates</identifier>
<label>Any action on autobackup templates</label>
</permission>
</permissions>
</role>
</roles>
</user>
</users>
<customer_network_id nil="true"/>
<disallowed_primary type="boolean">false</disallowed_primary>
<gateway>109.123.105.145</gateway>
<hypervisor_id nil="true"/>
<id type="integer">652</id>
<ip_address_pool_id nil="true"/>
<network_address>109.123.105.144</network_address>
<network_id type="integer">36</network_id>
<pxe type="boolean">false</pxe>
<updated_at type="datetime">2012-07-26T02:32:13-10:00</updated_at>
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>255.255.255.240</netmask>
</ip_address>
</used_ip_addresses>
<memory_available type="integer">3091</memory_available>
<disk_space_available type="integer">278</disk_space_available>
<cdn_reference type="integer">945453445</cdn_reference>
<additional_fields type="array">
<additional_field>
  <name>add_field1</name>
  <value>0</value>
</additional_field>
</additional_fields>
</user>
</users>

Where:

activated_at – time when the user was activated
avatar - user's avatar
bucket_id – ID of the bucket assigned to this user
cdn_account_status – always returns ACTIVE status; but it is actually activated when CDN was enabled for particular user
cdn_status - whether CDN resources are enabled for this user or not
created_at – time when the user was created, in [YYYY][MM][DD][hh][mm][ss]Z
deleted_at – time when the user was deleted
e-mail – user's e-mail
firewall_id - the ID of the firewall the user is associated with
first_name – user's first name
group_id - deprecated attribute; will be removed in upcoming release
id — the ID of a user in the database
image_template_group – the ID of associated template group, if any
infoboxes - an array of infoboxes with the following details:
  hidden_infoboxes - an array of hidden infoboxes
  hidden_infobox - hash of a hidden infobox
  display_infoboxes - true, if the infoboxes are displayed for this user, otherwise false
last_name – the user's last name
locale – locale (language) associated with user
login – user's login name
Login parameter is not returned when API key is used for authentication instead of a login/password combination.

`password_changed_at` - the date when the user's password was changed in the `[YYYY][MM][DD][hh][mm][ss]Z` format

`registered_yubikey` - true, if the user has enabled logging in using a YubiKey, otherwise false. To view the Yubikey(s) associated with a certain user, refer to Get List of User's YubiKeys. For the API request that adds a YubiKey, refer to Add YubiKey to User.

`status` – status of the user's account (active, suspended or deleted)

`supplied` - whether the user was created for the supplier when he published a zone to the Federation. If true, no actions can be performed on this user

`suspend_at` – time when the system should suspend a user

`system_theme` - color scheme of the interface: light or dark

`time_zone` – the time zone of the user

`total_amount` - sum total of outstanding and payment amount

`updated_at` – time when user's profile data was updated

`use_gravatar` – true, if avatar is enabled for the user, otherwise false

`user_group_id` – ID of the user group assigned to this user

`outstanding_amount` – the amount of money the user is due to pay

`payment_amount` – amount of money the user has actually paid

`roles` — an array of user roles to which this account is assigned to, where

  `created_at` - the date in the `[YYYY][MM][DD][hh][mm][ss]Z` format

  `id` – role ID

  `identifier` – role identifier

  `label` – role title

  `updated_at` - the date in the `[YYYY][MM][DD][hh][mm][ss]Z` format

`permissions`– an array with permissions assigned to this role

  `created_at` - the date in the `[YYYY][MM][DD][hh][mm][ss]Z` format

  `id` – permission ID

  `identifier` – permission identifier

  `label` – permission title

  `updated_at` - the date in the `[YYYY][MM][DD][hh][mm][ss]Z` format

`used_cpus` – number of CPU cores allocated to all virtual servers and edge servers of the user

`used_memory` - the amount of RAM used by the user (MB)

`used_cpu_shares` - the amount of CPU shares used

`used_disk_size` – size of all user disks in GB

`used_ip_addresses` – an array of IP addresses associated with the user

`ip_address` - an array of IP address with the following parameters:

  `address` - IP address

  `broadcast` – broadcast address

  `created_at` — the date in the `[YYYY][MM][DD][hh][mm][ss]Z` format
customer_network_id - the ID of the customer VLAN the IP address belongs to

disallowed_primary – true if not allowed to be used as primary (for VS build), otherwise false

gateway – gateway address

hypervisor_id - the ID of a compute resource the IP address is associated with

id – the ID of the IP address

ip_address_pool_id - the address of the IP address pool

network_address - the address of the network

network_id - the ID of the network

pxe - true, if this compute resource address can be used for cloudbooting a compute resource

updated_at – the date when the network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

user_id – ID of a user associated with this IP address

free – true if free, otherwise false

netmask – netmask for the IP address

memory_available – the amount of RAM available to this user (MB)

disk_space_available – disk space available for the user (GB)

additional field – user additional field, where:

value – the additional field value

name – the additional field title

86.4 Add User Group

To create a user group, use the following request:

POST /user_groups.xml
POST /user_groups.json

XML Request Example

curl -i -X POST http://onapp.test/user_groups.xml -d
  '<user_group><label>TEST_XML</label></user_group>' -u user:userpass -H
  'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X POST http://onapp.test/user_groups.json -d
  '{"user_group":{"label":"TEST_JSON"}}' -u user:userpass -H 'Accept:
  application/json' -H 'Content-type: application/json'

Where:

label* - the name of the user group

bucket_ids - IDs of buckets which will be available to users of this user group
86.5 Edit User Group

This API call allows you to edit a user group. It can also be used to assign a role and/or a billing plan to the user group.

To edit a user group, use the following request:

PUT /user_groups/:id.xml
PUT /user_groups/:id.json

XML Request Example

```
curl -i -X PUT http://onapp.test/user_groups/4.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<user_group><label>new_label</label><role_ids type="array"><role_id>2</role_id></role_ids><billing_plan_ids type="array"><billing_plan_id>1</billing_plan_id></billing_plan_ids></user_group>'
```

JSON Request Example

```
curl -i -X PUT http://onapp.test/user_groups/4.json -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"user_group":{"label":"new_label","role_ids":[2],"billing_plan_ids":[1]}}'
```

Where:

- **label** - choose user group name
- **role_id** - ID of the role(s) you wish to assign to the user group
- **bucket_id** - ID of the bucket(s) you wish to assign to the user group

86.6 Delete User Group

To delete a user group, use the following request:

DELETE /user_groups/:user_group_id.xml
DELETE /user_groups/:user_group_id.json

XML Request Example

```
```

JSON Request Example

```
```

Returns 204 response on successful deletion, or 404 response if no user group with such ID exists in the DB
87 Users

This class manages user accounts created in the cloud. It enables you to set up different types of user and allocate their role. Roles define user access to cloud resources and functions, including managing virtual servers and appliances, performing actions on templates and backups, and configuring data stores and networks.

87.1 Get List of Users

To see all the users registered in the cloud with their detailed information, use the following request:

GET /users.xml
GET /users.json

XML Request Example

```bash
curl -i -X GET -u user:password --url http://onapp.test/users.xml
```

JSON Request Example

```bash
curl -i -X GET -u user:password --url http://onapp.test/users.json
```

XML Output Example
<users type="array">
  <user>
    <activated_at type="datetime">2012-04-03T14:30:34+00:00</activated_at>
    <avatar nil="true"/>
    <cdn_account_status>ACTIVE</cdn_account_status>
    <cdn_status>ACTIVE</cdn_status>
    <created_at type="datetime">2012-04-03T14:30:34+00:00</created_at>
    <deleted_at nil="true"/>
    <email>admin@example.com</email>
    <firewall_id type="integer">4</firewall_id>
    <first_name>John</first_name>
    <group_id nil="true"/>
    <id type="integer">1</id>
    <identifier>cjximrxmt3yjbs</identifier>
    <infoboxes>
      <display_infoboxes type="boolean">false</display_infoboxes>
      <hidden_infoboxes type="array">
        <hidden_infobox>4840313084ef61f8e1dada293eb1b1ae</hidden_infobox>
      </hidden_infoboxes>
    </infoboxes>
    <last_name>Smith</last_name>
    <locale>en</locale>
    <login>admin</login>
    <password_changed_at type="datetime">2014-08-21T16:05:15+03:00</password_changed_at>
    <registered_yubikey>true</registered_yubikey>
    <status>active</status>
    <supplied type="boolean">false</supplied>
    <suspend_at nil="true"/>
    <system_theme>dark</system_theme>
    <time_zone>Baghdad</time_zone>
    <created_at type="datetime">2012-05-21T11:30:20+00:00</created_at>
    <user_group_id nil="true"/>
    <bucket_id type="integer">201</bucket_id>
    <used_cpus type="integer">0</used_cpus>
    <used_memory type="integer">0</used_memory>
    <used_cpu_shares type="integer">0</used_cpu_shares>
    <used_disk_size type="integer">0</used_disk_size>
    <memory_available type="integer">3091</memory_available>
    <disk_space_available type="integer">278</disk_space_available>
    <permissions type="array">
      <role>
        <created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
        <id type="integer">1</id>
        <label>Administrator</label>
        <updated_at type="datetime">2012-04-20T10:28:32+00:00</updated_at>
        <permissions type="array">
          <permission>
            <created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
            <id type="integer">110</id>
            <identifier>autobackup_templates</identifier>
            <updated_at type="datetime">2012-04-03T14:30:31+00:00</updated_at>
          </permission>
        </permissions>
      </role>
    </permissions>
  </user>
  <role>
    <created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
    <id type="integer">1</id>
    <identifier>admin</identifier>
    <label>Administrator</label>
    <updated_at type="datetime">2012-04-20T10:28:32+00:00</updated_at>
    <permissions type="array">
      <permission>
        <created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
        <id type="integer">110</id>
        <identifier>autobackup_templates</identifier>
        <updated_at type="datetime">2012-04-03T14:30:31+00:00</updated_at>
      </permission>
    </permissions>
  </role>
</users>
<additional_fields type="array">
  <additional_field>
    <name>add_field1</name>
    <value>0</value>
  </additional_field>
</additional_fields>

<used_ip_addresses type="array">
  <used_ip_address>
    <ip_address>
      <id type="integer">652</id>
      <address>109.123.105.147</address>
      <prefix type="integer">16</prefix>
      <broadcast>109.123.105.159</broadcast>
      <network_address>109.123.105.144</network_address>
      <gateway>109.123.105.145</gateway>
      <created_at type="datetime">2012-07-25T21:49:41-10:00</created_at>
      <updated_at type="datetime">2012-07-26T02:32:13-10:00</updated_at>
      <ipv4 type="boolean">true</ipv4>
      <hypervisor_id nil="true"/>
      <ip_range_id type="integer">1</ip_range_id>
    </ip_address>
  </used_ip_address>
</used_ip_addresses>

Where:

users - the array of users
user - the array of parameters for a user
activated_at – the date when the user was activated
avatar – the user's avatar
cdn_account_status – always returns ACTIVE status; but it is actually activated when CDN was enabled for particular user
cdn_status – whether CDN resources are enabled for this user or not
created_at – the date when the user was created, in [YYYY][MM][DD]T[hh][mm][ss]Z
deleted_at – the date when the user was deleted
email – the user's email
firewall_id – the ID of the firewall the user is associated with
first_name – the user's first name
group_id – the deprecated attribute; will be removed in upcoming release
id – the ID of a user in the database
identifier – the identifier of the user
image_template_group – the ID of associated template group, if any
infoboxes – the array of infoboxes with the following details:
  hidden_infoboxes – the array of hidden infoboxes
  hidden_infobox – the hash of a hidden inbox
display_infoboxes – true, if the infoboxes are displayed for this user, otherwise, false
last_name – the user's last name
locale – the locale (language) associated with user
login – the user's login name

Login parameter is not returned when API key is used for authentication instead of a login/password combination.

password_changed_at – the date when the user's password was changed in the [YYYY][MM][DD]T[hh][mm][ss]Z format
registered_yubikey – true, if the user has enabled logging in using a YubiKey, otherwise, false. To view the Yubikey(s) associated with a certain user, refer to Get List of User's YubiKeys. For the API request that adds a YubiKey, refer to Add YubiKey to User.
status – the status of the user's account (active, suspended, or deleted)
supplied – whether the user was created for the supplier when he published a zone to the Federation. If true, no actions can be performed on this user.
suspend_at – time when the system should suspend a user
system_theme – the color scheme of the interface: light or dark
time_zone – the time zone of the user
updated_at – time when user's profile data was updated
use_gravatar – true, if avatar is enabled for the user, otherwise false

user_group_id – ID of the user group assigned to this user
bucket_id – ID of the bucket assigned to this user
used_cpus – number of CPU cores allocated to all virtual servers and edge servers of the user
used_memory – the amount of RAM used by the user (MB)
used_cpu_shares – the amount of CPU shares used
used_disk_size – size of all user disks in GB
memory_available – the amount of RAM available to this user (MB)
disk_space_available – disk space available for the user (GB)
roles – an array of user roles to which this account is assigned to, where
created_at – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
id – the role ID

identifier – the role identifier
label – the role title
updated_at – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
permissions – an array with permissions assigned to this role
created_at – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
id – permission ID
identifier – permission identifier
updated_at – the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

monthly_price – the monthly fee for the bucket usage
payment_amount – amount of money the user has actually paid
outstanding_amount – the amount of money the user is due to pay
total_amount – sum total of outstanding and payment amount
discount_due_to_free – the price of the resources that were created within the buckets' free limits
total_amount_with_discount – the price of used resources that excludes the cost of the resources that were created within the bucket's free limits
used_ip_addresses – an array of IP addresses associated with the user
additional field – user additional field, where:
  value – the additional field value
  name – the additional field title
used_ip_addresses – an array of IP address with the following parameters:
  id – the ID of the IP address
  address – the IP address
  prefix – the prefix of the IP address
  broadcast – the broadcast address
  network_address – the address of the network
  gateway – the gateway address
  created_at – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
  updated_at – the date when the network was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
  network_id – the ID of the network
disallowed_primary – true if not allowed to be used as primary, otherwise false
customer_network_id – the ID of the customer VLAN the IP address belongs to
ipv4 – whether this is an IPv4 or and IPv6 IP address: ‘true’ for IPv4 IPs and ‘false’ for IPv6 IPs
user_id – the ID of a user associated with this IP address
hypervisor_id – the ID of a compute resource the IP address is associated with
ip_address_pool_id – ID of the IP address pool the IP address is associated with
ip_range_id – ID of the IP range the IP address is associated with
pxe – true, if this compute resource address can be used for a CloudBoot compute resource

Page History
v. 6.3 Edge 1
• removed the billing_plan_id parameter
v. 6.0
• added the billing_plan_id parameter for the backward compatibility with billing plans

87.2 Get List of Users (Short)
To view the list of users with short and fast output (without users' roles and permissions), use the following request:
GET /users.xml?short
GET /users.json?short

XML Request Example

```bash
curl -i -X GET -u user:password --url http://onapp.test/users.xml?short
```

JSON Request Example

```bash
curl -i -X GET -u user:password --url http://onapp.test/users.json?short
```

XML Output Example

```xml
<users type="array">
  <user>
    <activated_at type="datetime">2015-06-08T06:03:19+00:00</activated_at>
    <avatar nil="true"/>
    <cdn_account_status>ACTIVE</cdn_account_status>
    <cdn_status>INACTIVE</cdn_status>
    <created_at type="datetime">2015-06-08T06:03:18+00:00</created_at>
    <deleted_at nil="true"/>
    <email>admin@example.com</email>
    <firewall_id nil="true"/>
    <first_name>John</first_name>
    <group_id nil="true"/>
    <id type="integer">1</id>
    <identifier>e0ooz1qgipur9p</identifier>
    <infoboxes>
      <display_infoboxes type="boolean">true</display_infoboxes>
      <hidden_infoboxes type="array"/>
    </infoboxes>
    <last_name>Smith</last_name>
    <locale>en</locale>
    <login>admin</login>
    <password_changed_at type="datetime">2015-06-08T06:03:18+00:00</password_changed_at>
    <registered_yubikey type="boolean">false</registered_yubikey>
    <status>active</status>
    <supplied type="boolean">false</supplied>
    <suspend_at nil="true"/>
    <system_theme/>
    <time_zone>Kyiv</time_zone>
    <updated_at type="datetime">2015-06-10T10:39:34+00:00</updated_at>
    <use_gravatar nil="true"/>
    <user_group_id/>
    <bucket_id type="integer">7</bucket_id>
    <monthly_price type="decimal">0.0</monthly_price>
    <payment_amount type="decimal">0.0</payment_amount>
    <outstanding_amount type="decimal">0.0</outstanding_amount>
    <total_amount type="float">0.0</total_amount>
    <discount_due_to_free type="decimal">0.0</discount_due_to_free>
    <total_amount_with_discount type="decimal">0.0</total_amount_with_discount>
  </user>
</users>

Where:
users – the array of users
user – the array of parameters for the user
activated_at – the date when the user was activated
avatar – the user’s avatar

cdn_account_status – always returns ACTIVE status; but it is actually activated when CDN was enabled for a particular user
cdn_status – whether CDN resources are enabled for this user or not
created_at – the date when the user was created in the [YYYY][MM][DD]T[hh][mm][ss]Z format
deleted_at – the date when the user was deleted
email – the user’s email
firewall_id – the ID of the firewall the user is associated with
first_name – the user’s first name
group_id – the deprecated attribute that will be removed in upcoming release
id – the ID of a user in the database
identifier – the identifier of the user
image_template_group – the ID of the associated template group, if any
infoboxes – the array of infoboxes with the following details:
  hidden_infoboxes – the array of hidden infoboxes
  hidden_infobox – the hash of a hidden infobox
  display_infoboxes – true, if the infoboxes are displayed for this user, otherwise, false
last_name – the user’s last name
locale – the locale (language) associated with user
login – the user’s login name

The login parameter is not returned when an API key is used for authentication instead of a login/password combination.

password_changed_at – the date when the user’s password was changed in the [YYYY][MM][DD]T[hh][mm][ss]Z format
registered_yubikey – true, if the user has enabled logging in using a YubiKey, otherwise, false. To view the YubiKey(s) associated with a certain user, refer to Get List of User’s YubiKeys. For the API request that adds a YubiKey, refer to Add YubiKey to User.
status – the status of the user’s account (active, suspended, or deleted)
supplied – whether the user was created for the supplier when he published a zone to the Federation. If true, no actions can be performed on this user.
suspend_at – the date when the system can suspend a user in the [YYYY][MM][DD]T[hh][mm][ss]Z format
system_theme – the color scheme of the interface that can be light or dark
time_zone – the time zone of the user
updated_at – the date when the user’s profile data was updated
use_gravatar – true, if avatar is enabled for the user, otherwise, false
user_group_id – the ID of the user group assigned to this user
bucket_id – the ID of the bucket assigned to this user
monthly_price – the monthly fee for the bucket usage

payment_amount – the amount of money the user paid for consumed resources
outstanding_amount – the amount of money the user is due to pay
total_amount – the sum total of outstanding and payment amount
discount_due_to_free – the price of the resources that were created within the buckets' free limits
total_amount_with_discount – the price of used resources that excludes the cost of the resources that were created within the bucket's free limits

Page History
v. 6.3 Edge 1
• removed billing_plan_id parameter
v. 6.0
• added the billing_plan_id parameter for the backward compatibility with billing plans

87.3 Get User Details

To get details for a particular user account, use the following request:
GET /users/:id.xml
GET /users/:id.json

XML Request Example
```bash
curl -i -X GET -u user:password --url http://onapp.test/users/1.xml
```

JSON Request Example
```bash
curl -i -X GET -u user:password --url http://onapp.test/users/1.json
```

XML Output Example
<user>
<activated_at type="datetime">2012-04-03T14:30:34+00:00</activated_at>
<avatar nil="true"/>
<bucket_id type="integer">201</bucket_id>
<cdn_account_status>ACTIVE</cdn_account_status>
<cdn_status>ACTIVE</cdn_status>
<created_at type="datetime">2012-04-03T14:30:34+00:00</created_at>
<deleted_at nil="true"/>
<discount_due_to_free>0.0</discount_due_to_free>
<email>admin@example.com</email>
<firewall_id type="integer">4</firewall_id>
<first_name>John</first_name>
<group_id nil="true"/>
<id type="integer">1</id>
<image_template_group_id nil="true"/>
<infoboxes>
<hidden_infoboxes type="array">
<hidden_infobox>4840313084eef1f8e1dada293eb1b1ae</hidden_infobox>
</hidden_infoboxes>
</infoboxes>
<last_name>Smith</last_name>
<locale>en</locale>
<login>admin</login>
<monthly_price>0.0</monthly_price>
<password_changed_at type="datetime">2014-08-21T16:03:00+00:00</password_changed_at>
<registered_yubikey>true</registered_yubikey>
<status>active</status>
<supplied type="boolean">false</supplied>
<suspend_at nil="true"/>
<system_theme>dark</system_theme>
<time_zone/>
<total_amount type="float">83787.4296875</total_amount>
<total_amount_with_discount>526825.15</total_amount_with_discount>
<updated_at type="datetime">2012-05-21T11:30:20+00:00</updated_at>
<use_gravatar type="boolean">false</use_gravatar>
<user_group_id nil="true"/>
<outstanding_amount type="float">925.13000972271</outstanding_amount>
<payment_amount type="decimal">0.0</payment_amount>
<roles type="array">
<role>
<created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
<id type="integer">1</id>
<identifier>admin</identifier>
<label>Administrator</label>
<permissions type="array">
<permission>
<created_at type="datetime">2012-04-03T14:30:31+00:00</created_at>
<id type="integer">110</id>
<identifier>autobackup_templates</identifier>
<label>Any action on autobackup templates</label>
</permission>
</permissions>
</role>
</roles>
</user>
Where:

activated_at – time when the user was activated

avatar – user’s avatar

bucket_id – ID of the bucket assigned to this user

cdn_account_status – always returns ACTIVE status; but it is actually activated when CDN was enabled for particular user

cdn_status – whether CDN resources are enabled for this user or not

created_at – time when the user was created, in [YYYY][MM][DD][hh][mm][ss]Z

deleted_at – time when the user was deleted

discount_due_to_free – the price of the resources that were created within the buckets’ free limits.

e-mail – user’s email

firewall_id - the ID of the firewall the user is associated with

first_name – user’s first name

group_id - deprecated attribute; will be removed in upcoming release

id — the ID of a user in the database

image_template_group – the ID of associated template group, if any

infoboxes - an array of infoboxes with the following details:

    hidden_infoboxes - an array of hidden infoboxes

    hidden_infobox - hash of a hidden infobox

    display_infoboxes - true, if the infoboxes are displayed for this user, otherwise false

last_name – the user’s last name
locale – locale (language) associated with user
login – user’s login name

Login parameter is not returned when API key is used for authentication instead of a login/password combination.

monthly_price - monthly fee for bucket usage

password_changed_at - the date when the user’s password was changed in the [YYYY][MM][DD][hh][mm][ss]Z format

registered_yubikey - true, if the user has enabled logging in using a YubiKey, otherwise false. To view the Yubikey(s) associated with a certain user, refer to Get List of User's YubiKeys. For the API request that adds a YubiKey, refer to AddYubiKey to User.

status – status of the user’s account (active, suspended or deleted)
supplied - whether the user was created for the supplier when he published a zone to the Federation. If true, no actions can be performed on this user

suspend_at – time when the system should suspend a user

system_theme - color scheme of the interface: light or dark
time_zone – the time zone of the user
total_amount - sum total of outstanding and payment amount
total_amount_with_discount - the price of used resources that excludes the cost of the resources that were created within the bucket’s free limits.

updated_at – time when user's profile data was updated

use_gravatar – true, if avatar is enabled for the user, otherwise false

user_group_id – ID of the user group assigned to this user

outstanding_amount – the amount of money the user is due to pay

payment_amount – amount of money the user has actually paid

total_amount - sum total of outstanding and payment amount

total_amount_with_discount - the price of used resources that excludes the cost of the resources that were created within the bucket’s free limits.

updated_at – time when user's profile data was updated

use_gravatar – true, if avatar is enabled for the user, otherwise false

user_group_id – ID of the user group assigned to this user

outstanding_amount – the amount of money the user is due to pay

payment_amount – amount of money the user has actually paid

roles – an array of user roles to which this account is assigned to, where

created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

id – role ID

date - role identifier

label - role title

updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

permissions – an array with permissions assigned to this role

created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

id – permission ID

date - permission identifier

label – permission title

updated_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format

used_cpus – number of CPU cores allocated to all virtual servers and edge servers of the user
used_memory - the amount of RAM used by the user (MB)

used_cpu_shares - the amount of CPU shares used

used_disk_size – size of all user disks in GB

used_ip_addresses – an array of IP addresses associated with the user

ip_address - an array of IP address with the following parameters:

  address - IP address

  broadcast – broadcast address

  created_at — the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format

  customer_network_id - the ID of the customer VLAN the IP address belongs to

  disallowed_primary – true if not allowed to be used as primary (for VS build), otherwise false

  gateway – gateway address

  hypervisor_id - the ID of a compute resource the IP address is associated with

  id – the ID of the IP address

  ip_address_pool_id – the address of the IP address pool

  network_address – the address of the network

  network_id – the ID of the network

  pxe - true, if this compute resource address can be used for cloudbooting a compute resource

  updated_at – the date when the network was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format

  user_id – ID of a user associated with this IP address

  free – true if free, otherwise false

  netmask – netmask for the IP address

memory_available – the amount of RAM available to this user (MB)

disk_space_available – disk space available for the user (GB)

additional field – user additional field, where:

  value – the additional field value

  name – the additional field title

Page History

v.6.3 Edge 1

- removed billing_plan_id parameter

87.4 Check Login/E-mail Availability

To check the username availability, use the following request:

POST http://onapp.test/users/validate_login.xml
POST http://onapp.test/users/validate_login.json

XML Request Example
curl -i -X POST -u user:userpass -d '"login":"admin"'}

Where:

**login** – desired username

You can also use the following request type:

curl -X POST -u user:userpass
http://onapp.test/users/validate_login.json?login=admin

To check the e-mail availability, use the following request:

**XML Request Example**

curl -i -X POST -u user:userpass -d '"email":"sfdsf@dg.yu"'}

**JSON Request Example**

curl -i -X POST -u user:userpass -d '"email":"sfdsf@dg.yu"'}

Where:

**email**– desired username

**XML Output Example**

If the username is available:
If the username is not available:

```json
{"valid":false,"message":"Username has already been taken"}

<user>
  <valid type="boolean">false</valid>
  <message>Username has already been taken</message>
</user>
```

87.5 Add User

To create a new user account, use the following request:

**POST /users.xml**
**POST /users.json**

**XML Request Example**

```bash
curl -i -X POST -d '<user><login>111111losj</login><email>MailTestApi@testmatil.com</email><first_name>TestApiName</first_name><last_name>TestAPIName</last_name><password>password_test1</password><user_group_id>1</user_group_id><role_ids type="array"><role_id>1</role_id></role_ids><additional_fields type="array"><additional_field><name>additional_field_name</name><value>custom_value</value></additional_field></additional_fields><time_zone>Kyiv</time_zone><locale>en</locale></user>' -u user:userpass http://onapp.test/users.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```bash
curl -i -X POST -d '{"user":{"login":"111111losj","email":"111111losj@test.test","first_name ":"1111","last_name":"1311","password":"password_test1","user_group_id":"1 ","bucket_id":"1","role_ids":["1","2"],"additional_fields":{"additional_field":{"name":"additional_field_name"," value":"custom_value"}}}}' -u user:userpass http://onapp.test/users.json -H 'Accept: application/json' -H 'Content-type: application/json'
```

Returns HTTP 201 on successful creation, or HTTP 422 if a user with such a login/email already exists.

**Where:**

- **email** - user's email address
- **login** - login of the user. It can consist of 2-40 characters, letters [A-Za-z], digits [0-9], dash [-], lower dash [ _ ], [@]. You can use both lower- and uppercase letters. The dash [- ] and [@] symbols are not allowed as first characters of the login name.
password* - user’s password. (min – 6 characters)

first_name - user’s first name

last_name - user’s last name

role – assigns a role to a user

time_zone - time zone of the user. Set by default

locale - local of the user. Set by default

status – user’s status (active, suspended, etc)

bucket_id – set by default, if not selected

additional field – an array of custom fields assigned to the user, where:
- name - the name of a particular additional field
- value - the value which you want to assign to this additional field

role_ids – an array of the role IDs, assigned to the user

user_group_id – ID of the group, to which the user is attached

suspend_after_hours – time in hours, after which the user will be suspended

suspend_at – time in [YYYY][MM][DD] T[hh][mm][ss]Z format, when the user will be suspended

Page History

v. 4.0
- the login length decreased to two symbols

v. 3.1
- first_name and last_name are no longer required parameters

87.6 Edit User

To edit a user, use the following request:

PUT /users/:id.xml
PUT /users/:id.json

XML Request Example

```
curl -i -X PUT -d
  
  '<user><email>somemail@example.com</email><first_name>NewName</first_name>
  <last_name>NewLastName</last_name><password>qwe123</password><user_group_i
  d>36</user_group_id><bucket_id>2</bucket_id><role_ids
  type="array"><role_id>1</role_id></role_ids><additional_fields
type="array"><additional_field><name>additional_field_name</name><value>cu
  stom_value</value></additional_field></additional_fields><suspend_at>2011-08-01
  12:47:08</suspend_at><registered_yubikey>true</registered_yubikey></user>'
  -u user:userpass http://onapp.test/users/12.xml -H 'Accept:
  application/xml' -H 'Content-Type: application/xml'
```

JSON Request Example
\{'"user":{"email":"1111111losj@test.test","first_name":"1111","last_name":"1311","password":"changeme","user_group_id":"1","bucket_id":"1","role_ids ":["1","2"],\\n"additional_fields":\[\"additional_field":\{"name":"additional_field_name","value":"custom_value"\}\]\}\'}

Where:

email - user's email address
first_name - user's first name
last_name - user's last name
password - user's password. (min – 6 characters)
user_group_id - ID of the group, to which the user is attached
bucket_id - ID of the bucket assigned to this user
role_ids - an array of the role IDs, assigned to the user
additional field – an array of custom fields assigned to the user, where:
  • name - the name of a particular additional field
  • value - the value which you want to assign to this additional field
suspend_at - time in [YYYY][MM][DD] T[hh][mm][ss]Z format, when the user will be suspended
registered_yubikey - set to 'true' if you want to enable logging in using a Yubikey for this user, otherwise set to 'false'. For the API request that adds a YubiKey, refer to Add YubiKey to User.

• To disable user auto-suspending, leave the suspend_at field empty.
• Users with API key instead of password are not allowed to see their login and change their password.

Page History
v. 4.2
• added the registered_yubikey parameter

87.7 Suspend User

To suspend a user account, use the following request:

POST /users/:id/suspend.xml
POST /users/:id/suspend.json

XML Request Example

87.8 Activate User

To activate a suspended user account, use the following request:

POST /users/:id/activate.xml
POST /users/:id/activate.json

XML Request Example

```
```

JSON Request Example

```
```

87.9 Unlock User

If your account has been locked because of exceeded number of unsuccessful login attempts, you may unlock it. To unlock an account, use the following request:

POST http://onapp.test/users/:id/unlock_account.xml
POST http://onapp.test/users/:id/unlock_account.json

XML Request Example

```
```

JSON Request Example

```
```

Where:

unlock_token - unlock token that will be sent to the user email address.
87.10 Delete User

To remove a user account from the cloud, use the following request:

DELETE /users/:id.xml
DELETE /users/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Returns HTTP 204 response on successful deletion, or HTTP 404 when a user with the ID specified is not found.

When you delete a user their status becomes DELETED and their backups, virtual servers, load balancers and CDN resources will be marked as removed. To completely remove a user and their billing statistics from the system, run DELETE /users/:id again.

To delete a user along with their backups, load balancers and CDN resources with a single API call, run:

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

87.11 Drop Sessions

To terminate all active sessions, use the following request:

DELETE http://onapp.test/users/drop_all.xml
DELETE http://onapp.test/users/drop_all.json

**XML Request Example:**

**JSON Request Example:**


### 87.12 Get User Last Access Log

To view user's last access log details, use the following request:

GET http://onapp.test/users/:user_id/last_access_log.xml
GET http://onapp.test/users/:user_id/last_access_log.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**
<last_access_log_items type="array">
    <last_access_log_item>
        <created_at type="datetime">2013-09-24T11:24:50+03:00</created_at>
        <id type="integer">1</id>
        <ip>98.158.179.190</ip>
        <user_id type="integer">333</user_id>
    </last_access_log_item>
    <last_access_log_item>
        <created_at type="datetime">2013-09-24T11:26:11+03:00</created_at>
        <id type="integer">2</id>
        <ip>98.158.179.190</ip>
        <user_id type="integer">333</user_id>
    </last_access_log_item>
    <last_access_log_item>
        <created_at type="datetime">2013-09-24T12:20:53+03:00</created_at>
        <id type="integer">3</id>
        <ip>194.44.160.178</ip>
        <user_id type="integer">617</user_id>
    </last_access_log_item>
    <last_access_log_item>
        <created_at type="datetime">2013-09-24T12:27:46+03:00</created_at>
        <id type="integer">4</id>
        <ip>194.44.160.178</ip>
        <user_id type="integer">1873</user_id>
    </last_access_log_item>
    <last_access_log_item>
        <created_at type="datetime">2013-09-24T12:27:59+03:00</created_at>
        <id type="integer">5</id>
        <ip>98.158.179.190</ip>
        <user_id type="integer">333</user_id>
    </last_access_log_item>
</last_access_log_items>

Where:
- **id** - log's ID
- **user_id** - user's ID
- **ip** - IP address

### 87.13 Get User Statistics

User’s statistics show the summary of the resources used by a particular user and their costs (which are set in the bucket assigned to the user). To get the user statistics, use the following request:

GET /users/:user_id/user_statistics.xml
GET /users/:user_id/user_statistics.json

**XML Request Example**


**JSON Request Example**

XML Output Example
```xml
<user_stat>
<vm_stats type="array">
<vm_stat>
<virtual_machine_id type="integer">5551</virtual_machine_id>
<total_cost type="decimal">0.0</total_cost>
</vm_stat>
...
</vm_stats>
<stat_time type="dateTime">2019-05-29T13:03:30Z</stat_time>
<vm_cost type="decimal">1276.0</vm_cost>
<user_resources_cost type="decimal">842633.043828097</user_resources_cost>
<user_resources_discount_due_to_free type="decimal">53801.043828097</user_resources_discount_due_to_free>
<total_cost type="decimal">843909.043828097</total_cost>
<monthly_discount_due_to_free type="decimal">0.692663813</monthly_discount_due_to_free>
<total_discount_due_to_free type="decimal">53961.73649191</total_discount_due_to_free>
<total_cost_with_discount type="decimal">789947.307336187</total_cost_with_discount>
<currency_code>USD</currency_code>
<user_id type="integer">5</user_id>
<backup_cost type="decimal">0.0</backup_cost>
<backup_count_cost type="decimal">108.0</backup_count_cost>
<template_cost type="decimal">13.0</template_cost>
<template_count_cost type="decimal">110.0</template_count_cost>
<storage_disk_size_cost type="decimal">137.6707077</storage_disk_size_cost>
<storage_disk_size_discount_due_to_free type="decimal">137.6707077</storage_disk_size_discount_due_to_free>
<backup_disk_size_cost type="decimal">66.81032182</backup_disk_size_cost>
<backup_disk_size_discount_due_to_free type="decimal">66.81032182</backup_disk_size_discount_due_to_free>
<template_count_discount_due_to_free type="decimal">104.0</template_count_discount_due_to_free>
<template_disk_size_cost type="decimal">484.366775531</template_disk_size_cost>
<template_disk_size_discount_due_to_free type="decimal">484.366775531</template_disk_size_discount_due_to_free>
<recovery_point_cost type="integer">0</recovery_point_cost>
<recovery_point_size_cost type="integer">0</recovery_point_size_cost>
<autoscale_cost type="integer">0</autoscale_cost>
<acceleration_cost type="integer">0</acceleration_cost>
<ova_count_cost type="integer">0</ova_count_cost>
```

<ova_size_cost type="integer">0</ova_size_cost>
<ova_size_discount_due_to_free type="integer">0</ova_size_discount_due_to_free>
<edge_group_cost type="integer">0</edge_group_cost>
<backup_resource_usage_cost type="integer">0</backup_resource_usage_cost>
<backup_resource_usage_discount_due_to_free type="integer">0</backup_resource_usage_discount_due_to_free>
</user_stat>

Where:

vm_stat – billing statistics on virtual servers, owned by the user
virtual_machine_id – ID of the VS, for which this statistics is generated
user_resources_cost – costs for actual usage of the VS
backup_disk_size_cost - cost for the backups of a particular Backup server zone. The price is set by the Limits for backup server zone.
backup_cost – cost per backup per particular point of time for which these statistics are generated
backup_count_cost - price per backups located on backup server
template_iso_cost – cost per ISO per hour
autoscale_cost – cost per number of VSs using Autoscaling
customer_network_cost – cost per number of customer networks
acceleration_cost – cost per number of accelerated VSs
ova_cost – cost per OVA file per hour
ova_size_cost – cost per GB per hour for OVA file disk size
currency_code - the currency set for this user
template_disk_size_cost - cost for the templates of a particular Backup server zone
template_cost — total template costs on compute resources (cost per template per hour)
edge_group_cost - total edge group costs
user_id - the ID of the user for whom the statistics are generated
stat_time - a particular point of time for which these statistics are generated
storage_disk_size_cost – costs for disk size used for backups/templates storage (cost per GB per hour)
user_resources_cost — sum total of all backups/templates/monitis monitors costs (backup_cost+storage_disk_size_cost+template_cost+monit_cost)
total_cost — overall sum total of all costs (vm_cost+user_resources_cost)
vm_cost – total VSs costs (sum total of all user VSs)
vm_discount_due_to_free - VS discount counted based on free limits in a rate card
user_resources_discount_due_to_free - user resources discount counted based on free limits in a rate card
moonthly_discount_due_to_free - on a monthly basis (including monthly peak free limits+ monthly free limits)
total_discount_due_to_free - sum of all discounts counted based on free limits in a rate card
total_cost_with_discount - total_cost value received after subtracting the total_discount_due_to_free value
backup_discount_due_to_free - backup discount counted based on free limits in a rate card

template_discount_due_to_free - template discount counted based on free limits in a rate card

template_iso_discount_due_to_free - ISO discount counted based on free limits in a rate card

storage_disk_size_discount_due_to_free - discount for disk size used for backups/templates storage

backup_disk_size_discount_due_to_free - cost for the backup disks of a particular Backup

template_count_cost - price for templates located on backup server

template_count_discount_due_to_free - template discount counted based on free limits in a rate card

template_disk_size_discount_due_to_free - discount for disk size used for templates located on backup server

recovery_point_cost - cost per recovery point of the backup resource

recovery_point_discount_due_to_free - recovery point discount counted based on free limits in a rate card

recovery_point_size_cost - cost per backup size on backup resource

recovery_point_size_discount_due_to_free - discount for recover point size used for templates located on backup server

autoscale_discount_due_to_free - discount for VSs using autoscaling counted based on free limits in a rate card

acceleration_discount_due_to_free - discount for accelerated VSs counted based on free limits in a rate card

ova_count_cost - cost per ova template

ova_size_discount_due_to_free - discount for disk size used for OVAs located on backup server

backup_resource_usage_cost - cost per used backup size

backup_resource_usage_discount_due_to_free - backup resource usage discount counted based on free limits in a rate card

Page History

v.5.6

- removed the vm_resource_cost parameter
- replaced the backup_count_cost parameter with the backup_cost parameter
- replaced the ova_count_cost parameter with the ova_cost parameter
- replaced the template_count_cost parameter with the template_cost parameter
- added the following parameters:
  - vm_discount_due_to_free
  - user_resources_discount_due_to_free
  - monthly_discount_due_to_free
  - total_discount_due_to_free
  - total_cost_with_discount
  - backup_discount_due_to_free
  - template_discount_due_to_free
  - template_iso_discount_due_to_free
v.5.2
  • added the following parameters:
    o template_iso_cost
    o autoscale_cost
    o customer_network_cost
    o acceleration_cost
    o ova_count_cost
    o ova_size_discount_due_to_free
  o backup_resource_usage_discount_due_to_free

87.14 Get User’s Statistics for Particular Period

To view the hourly cost and amount of the resources used by a user, use the following request:

GET /users/:user_id/user_statistics.xml?hourly_stats
GET /users/:user_id/user_statistics.json?hourly_stats

XML Request Example:

curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass

JSON Request Example:

curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass

XML Output Example:
<user_stats type="array">
  <user_stat>
    <backup_cost type="decimal">0.0</backup_cost>
    <user_resources_cost type="decimal">0.0</user_resources_cost>
    <currency_code>USD</currency_code>
    <total_cost type="decimal">0.0</total_cost>
    <template_cost type="decimal">0.0</template_cost>
    <storage_disk_size_cost type="decimal">0.0</storage_disk_size_cost>
    <edge_group_cost type="decimal">0.0</edge_group_cost>
    <backup_cost type="decimal">0.0</backup_cost>
    <template_cost type="decimal">0.0</template_cost>
    <user_id type="integer">4</user_id>
    <backup_disk_size_cost type="decimal">0.0</backup_disk_size_cost>
    <monit_cost type="decimal">0.0</monit_cost>
    <stat_time type="datetime">2012-03-13T11:43:36Z</stat_time>
    <template_disk_size_cost type="decimal">0.0</template_disk_size_cost>
    <vm_cost type="integer">0</vm_cost>
    <vm_stats type="array"/>
  </user_stat>
  ...
</user_stats>

For parameters description, refer to View user's statistics section.

To view the cost and amount of the resources used for a particular period:

GET
/users/:user_id/user_statistics.xml?hourly_stats&period[startdate]=YYYY-MM-DD+hh%3Amm%3Ass&period[enddate]=YYYY-MM-DD+hh%3Amm%3Ass
GET
/users/:user_id/user_statistics.json?hourly_stats&period[startdate]=YYYY-MM-DD+hh%3Amm%3Ass&period[enddate]=YYYY-MM-DD+hh%3Amm%3Ass

XML Request Example

curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass

JSON Request example

XML Output Example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<user_stats type="array">
  <user_stat>
    <backup_cost type="decimal">0.0</backup_cost>
    <user_resources_cost type="decimal">0.0</user_resources_cost>
    <currency_code>USD</currency_code>
    <total_cost type="decimal">0.0</total_cost>
    <template_cost type="decimal">0.0</template_cost>
    <storage_disk_size_cost type="decimal">0.0</storage_disk_size_cost>
    <edge_group_cost type="decimal">0.0</edge_group_cost>
    <backup_cost type="decimal">0.0</backup_cost>
    <template_cost type="decimal">0.0</template_cost>
    <user_id type="integer">4</user_id>
    <backup_disk_size_cost type="decimal">0.0</backup_disk_size_cost>
    <monit_cost type="decimal">0.0</monit_cost>
    <stat_time type="datetime">2012-03-15T07:58:17Z</stat_time>
    <template_disk_size_cost type="decimal">0.0</template_disk_size_cost>
    <vm_cost type="integer">0</vm_cost>
    <vm_stats type="array"/>
  </user_stat>
  <user_stat>
    <backup_cost type="decimal">0.0</backup_cost>
    <user_resources_cost type="decimal">0.0</user_resources_cost>
    <currency_code>USD</currency_code>
    <total_cost type="decimal">0.0</total_cost>
    <template_cost type="decimal">0.0</template_cost>
    <storage_disk_size_cost type="decimal">0.0</storage_disk_size_cost>
    <edge_group_cost type="decimal">0.0</edge_group_cost>
    <backup_cost type="decimal">0.0</backup_cost>
    <template_cost type="decimal">0.0</template_cost>
    <user_id type="integer">4</user_id>
    <backup_disk_size_cost type="decimal">0.0</backup_disk_size_cost>
    <monit_cost type="decimal">0.0</monit_cost>
    <stat_time type="datetime">2012-03-15T07:58:17Z</stat_time>
    <template_disk_size_cost type="decimal">0.0</template_disk_size_cost>
    <vm_cost type="integer">0</vm_cost>
    <vm_stats type="array"/>
  </user_stat>
</user_stats>
```

For parameters description, refer to View user's statistics section.

Page History

v.5.6

- replaced the backup_count_cost parameter with the backup_cost parameter
- replaced the ova_count_cost parameter with the ova_cost parameter
- replaced the template_count_cost parameter with the template_cost parameter
87.15 Get User Billing Statistics

To view billing statistics for a particular user, use the following request:

GET /users/:user_id/vm_stats.xml
GET /users/:user_id/vm_stats.json

To get a shorter statistics output, add an id parameter to the URL:

GET /users/:user_id/vm_stats/:vm_stats_id.xml
GET /users/:user_id/vm_stats/:vm_stats_id.json

If the account was created less than three months ago, the statistics are generated for the entire period of operation of the account. You can also define a shorter period by setting Start and End time in the API call:

GET /users/:user_id/vm_stats.xml?period[startdate]=YYYY-MM-DD+hh%3Amm%3Ass&period[enddate]=YYYY-MM-DD+hh%3Amm%3Ass
GET /users/:user_id/vm_stats.json?period[startdate]=YYYY-MM-DD+hh%3Amm%3Ass&period[enddate]=YYYY-MM-DD+hh%3Amm%3Ass

XML Output Example
<vm_stats type="array">
  <vm_hourly_stat>
    <created_at type="datetime">2013-05-01T00:00:28Z</created_at>
    <currency_code>USD</currency_code>
    <id type="integer">13888</id>
    <stat_time type="datetime">2013-05-01T00:00:00Z</stat_time>
    <updated_at type="datetime">2013-05-01T00:00:28Z</updated_at>
    <user_id type="Integer">239</user_id>
    <virtual_machine_id nil="true"/>
    <vm_billing_stat_id type="integer">7246</vm Billing_stat_id>
    <total_cost type="decimal">0.0</total_cost>
    <usage_cost type="decimal">0.0</usage_cost>
  </vm_hourly_stat>
  <vm_hourly_stat>
    <created_at type="datetime">2013-05-01T01:00:27Z</created_at>
    <currency_code>USD</currency_code>
    <id type="integer">13937</id>
    <stat_time type="datetime">2013-05-01T01:00:00Z</stat_time>
    <updated_at type="datetime">2013-05-01T01:00:27Z</updated_at>
    <user_id type="integer">239</user_id>
    <virtual_machine_id nil="true"/>
    <vm_billing_stat_id type="integer">7246</vm Billing_stat_id>
    <total_cost type="decimal">0.0</total_cost>
    <usage_cost type="decimal">0.0</usage_cost>
  </vm_hourly_stat>
  <vm_stats type="array">

Where:

created_at – the timestamp in DB when this record was created
updated_at – the time stamp in DB when this record was updated
currency_code - currency in which this virtual server is charged within the bucket
id – the ID of the server hourly statistics. You can add this parameter to the request URL to get a shorter statistics output.
stat_time – the particular hour for which these statistics were generated
user_id - the ID of VS owner
virtual_machine_id - ID of a virtual server
virtual_machine_billing_statistics_id - ID of a virtual server billing statistics
billing_stats - an array of billing details for the resources used by this VS:

The generated billing statistics will show the billing details for all virtual servers, load balancers, edge servers and storage servers owned by this particular user. For the output examples and the explanation of the fields returned, refer to corresponding sections:

- Get VS Billing Statistics
- Load Balancer Billing Statistics
- CDN Edge Server Billing Statistics
- Get CDN Storage Server Billing Statistics

Page History

v.5.6
removed the \texttt{vm\_resources\_cost} parameter

### 87.16 Get List of User Monthly Bills

To get data on user's monthly bills for a year, use the following request:

GET \texttt{/users/:user\_id/monthly\_bills.xml}
GET \texttt{/users/:user\_id/monthly\_bills.json}

To view user monthly bills for a particular year, use the following request:

**XML Request Example**

```bash
curl -X GET -u 'user:userpass'
https://onapp.test/users/12/monthly_bills.xml -d "date[year]=2015"
```

**JSON Request Example**

```bash
curl -X GET -u 'user:userpass'
https://onapp.test/users/12/monthly_bills.json -d "date[year]=2015"
```

**XML Output Example**

```
<vm_stats type="array">
  <vm_stat>
    <month type="integer">5</month>
    <cost type="float">167.371330738068</cost>
  </vm_stat>
</vm_stats>
```

Where:

- \textit{month} - number of a month
- \textit{cost} - total user costs, charged for that month (monthly price+costs for resources and usage. See the \texttt{Get User Billing Statistics} section for details)

### 87.17 Get List of User Payments

To get the list of user payments, use the following request:

GET \texttt{/users/:user\_id/payments.xml}
GET \texttt{/users/:user\_id/payments.json}

**XML Request Example**

```bash
curl -X GET -u 'user:userpass'
http://onapp.test/users/12/payments.xml
```

**JSON Request Example**

---
87.18 Add Payment

To add a payment record to your DB, use the following request:

Be aware, that starting with OnApp 5.4 version, the request below will be deprecated. Instead, only the Create User Payment request will be used.

POST /users/:user_id/payments.xml
POST /users/:user_id/payments.json

Where:

amount — money amount in the currency set in the bucket
invoice_number — optional number of invoice
id — payment ID
user_id — ID of the user
**amount** - amount of the payment (should be higher than zero)

**invoice_number** - optional number of the invoice

### 87.19 Edit Payment

To change the *invoice number* or the payment *amount*, use the following request:

Be aware, that starting with OnApp 5.4 version, the request below will be deprecated. Instead, only the [Edit User Payment](#) request will be used.

**XML Request Example**

```bash
curl -i -X PUT -d 
  '<?xml version="1.0" encoding="UTF-8"?>
  <payment>
    <amount>99</amount>
    <invoice_number>66</invoice_number>
  </payment>'
  -H 'Accept: application/xml' -H 'Content-type: application/xml'
  -u user:userpass -url http://onapp.test/users/8/payments/3.xml
```

**JSON Request Example**

```bash
curl -i -X PUT -d 
  '{
   "payment": {
       "amount": "99",
       "invoice_number": "66"
   }
  }' -H
  'Accept: application/json' -H 'Content-type: application/json'
  -u user:userpass -url http://onapp.test/users/8/payments/3.json
```

Where:

**amount** - amount of the payment (should be higher than zero)

**invoice_number** - optional number of the invoice

### 87.20 Delete Payment

Be aware, that starting with OnApp 5.4 version, the request below will be deprecated. Instead, only the [Delete User Payment](#) request will be used.

**XML Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/users/3/payments/2.xml
```

**JSON Request Example**

```bash
curl -i -X DELETE -u user:userpass --url http://onapp.test/users/3/payments/2.json
```
87.21 Get List of User VSs

To see the virtual servers owned by a particular user, use the following request:

GET /users/:user_id/virtual_machines.xml
GET /users/:user_id/virtual_machines.json

XML Request Example

```
```

JSON Request Example

```
```

An array of virtual servers will be returned.

For details and parameter description, refer to Get the list of VSs section.

87.22 Get List of Compute Resources Used by Users' VSs

To get the list of compute resources used by user's VSs, use the following request:

GET onapp.com/users/:user_id/hypervisors.xml
GET onapp.com/users/:user_id/hypervisors.json

XML Request Example

```
```

JSON Request Example

```
```

An array of compute resources used by VSs of the user will be returned.

For details and parameters description refer to Get List of Compute Resources section.
87.23 Get List of User Backups

To view backups of a particular user, use the following request:

GET /users/:user_id/backups.xml
GET /users/:user_id/backups.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

For output example and parameters description, refer to Get the List of VS Backups section.

87.24 Get List of User Data Store Zones

To see data store zones assigned to user's VSs, use the following request:

GET /users/:user_id/data_store_zones.xml
GET /users/:user_id/data_store_zones.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

For details and parameters description refer to Get List of Data Store Zones section.

87.25 Get List of User Limits

Limits display available resources for creating a VS, but not all the available resources of the user.

To see user limits, use the following request:

GET /users/:user_id/limits.xml
GET /users/:user_id/limits.json

XML Request Example

```bash
```

JSON Request Example


XML Output Example
<hash>
<limits>
<memory type="integer">12761</memory>
<cpus type="integer">8</cpus>
<cpu-shares type="integer">800</cpu-shares>
<rate type="integer">1000</rate>
<hypervisor_groups type="array">
<hypervisor_group>
$id type="integer">2</id>
<label>QA HVZ</label>
</hypervisor_group>
</hypervisor_groups>
<hypervisors type="array">
<hypervisor>
$id type="integer">1</id>
<label>HV2_xen</label>
</hypervisor>
<hypervisor>
$id type="integer">12</id>
<label>HVL-xen</label>
</hypervisor>
<hypervisor>
$id type="integer">13</id>
<label>KVM_HV2</label>
</hypervisor>
</hypervisors>
<data_store_groups type="array">
<data_store_group>
$id type="integer">3</id>
<label>dsz</label>
<data_stores type="array">
<data_store>
$id type="integer">1</id>
<usage type="integer">188</usage>
<capacity type="integer">460</capacity>
</data_store>
<data_store>
$id type="integer">22</id>
<usage type="integer">154</usage>
<capacity type="integer">1800</capacity>
</data_store>
</data_stores>
</data_store_group>
</data_store_groups>
<network_groups type="array">
<network_group>
$id type="integer">4</id>
<label>ntz6-xen</label>
</network_group>
<network_group>
$id type="integer">29</id>
<label>0Hnz2</label>
</network_group>
<network_group>
<primary-disk-size type="integer">1646</primary-disk-size>
<swap-disk-size type="integer">1646</swap-disk-size>
</limits>
<best-data-store-group-primary-id type="integer">3</best-data-store-group-
primary-id>
<best-data-store-group-swap-id type="integer">3</best-data-store-group-
swap-id>
<best-network-group-id type="integer">29</best-network-group-id>
</hash>
Where:

- **cpu** – amount of CPU cores, available for the user to create a VS
- **cpu_shares** - CPU priority available for creation a VS
- **swap_disk_size/primary_disk_size** – available disk space in GB at best_data_store_group_primary_id (best_data_store_group_swap_id)
- **memory** – available RAM
- **rate** – maximum port speed limit
- **hypervisor_groups** – an array of available compute zones, with zone label and ID
- **hypervisors** – an array of available compute resources, with compute resource label and ID
- **data_store_groups** – an array of available data store groups, with group label, ID and the list of assigned data stores.
- **data_store** – an array of data stores assigned to the data store groups, with ID, usage and capacity
- **network_group** – an array of network groups, with group label and ID
- **best_data_store_group_primary_id(best_data_store_group_swap_id)** – the ID of a data store zone with higher available disk capacity.
- **best_network_group_id** – the ID of a priority network

87.26 Get List of User Network Zones

To get the list of network zones associated with a user:

GET /users/:user_id/network_zones.xml
GET /users/:user_id/network_zones.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

For details and parameters description refer to [Get List of Network Zones](#) section.

87.27 Search User Backups

To find a backup of a particular user, use the following request:

GET /users/:user_id/backups_search.xml
GET /users/:user_id/backups_search.json

**XML Request Example**

```bash
```
JSON Request Example

```
curl -X GET -u user:userpass
```

Where:

- **size** – set the size [from] and [to] backup search parameters to search backups by their size. The size should be indicated in MB.
- **date** – set the date [startdate] and [enddate] backup search parameters to search for backups created between two dates. The date should be indicated in the YYYY-MM-DD format.

Please be aware that some Unix command shells can output an error because of square brackets. To prevent the error, use the back slash escape symbol. The example curl with back slashes is as follows:

XML Request Example

```
curl -X GET -u user:userpass
```

XML Output Example

```
curl -X GET -u user:userpass
```
<backups type="array">
  <backup>
    <allow_resize_without_reboot type="boolean">true</allow_resize_without_reboot>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <backup_server_id type="integer">1</backup_server_id>
    <built type="boolean">true</built>
    <built_at type="datetime">2016-09-21T12:26:11Z</built_at>
    <created_at type="datetime">2016-09-21T12:08:06Z</created_at>
    <data_store_type>lvm</data_store_type>
    <id type="integer">872</id>
    <identifier>dyhy150m</identifier>
    <initiated>manual</initiated>
    <iqn nil="true"/>
    <locked type="boolean">false</locked>
    <marked_for_delete type="boolean">false</marked_for_delete>
    <min_disk_size type="integer">9</min_disk_size>
    <min_memory_size type="integer">384</min_memory_size>
    <note>zaza50patch76</note>
    <operating_system>linux</operating_system>
    <operating_system_distro>rhel</operating_system_distro>
    <target_id type="integer">9287</target_id>
    <template_id type="integer">28</template_id>
    <updated_at type="datetime">2016-09-21T12:26:15Z</updated_at>
    <user_id type="integer">3</user_id>
    <volume_id nil="true"/>
    <backup_type>normal</backup_type>
    <disk_id type="integer">9287</disk_id>
  </backup>
  ...</backup>
</backups>

Where:

allow_resize_without_reboot - true if the template to which the backup can be restored will support resize without reboot option, otherwise false

allowed_hot_migrate - true if the template to which the backup can be restored will support hot migration, otherwise false.

allowed_swap - true if the template to which the backup can be restored will allow swap, otherwise false.

backup_server_id - the ID of the backup server on which the backup is stored.

backup_size - the size of the backup

built - true if the backup is already built, otherwise false

built_at - the date in the [YYYY][MM][DD][T][hh][mm][ss][Z] format

created_at - the date in the [YYYY][MM][DD][T][hh][mm][ss][Z] format

data_store_type - data store type: lvm, vmware or solidfire

id - the backup ID

identifier - the backup identifier

initiated - the period when a backup is initiated: days, weeks, months, or years

iqn - volume ISCSI qualified name (SolidFire-related parameter)

locked - true if the backup is being built, otherwise false

marked_for_delete - the backup is marked for deletion (for auto-backups)
**min_disk_size** - minimum disk size required for restoring a backup

**min_memory_size** - minimum memory size required for restoring a backup

**note** - an optional note to the backup

**operating_system** - the OS of the VS from which the backup was created

**operating_system_distro** - the OS distribution of the VS from which the backup was created

**target_id** - ID of a backup target

**target_type** - the target for which the backup was taken; For normal backups, it is a disk. For incremental backups, it's a virtual server.

**template_id** - the ID of the template the VS is based on

**updated_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format

**user_id** - the ID of a user the storage server belongs to

**volume_id** - data store ID

**backup_type** - disk backup type (normal, Days auto-backup, Weeks auto-backup, Months auto-backup, Years auto-backup)

**disk_id** - the ID of the backed up disk

### 87.28 Generate API Key

To generate a new API key, use the following request:

POST /users/:id/make_new_api_key.xml

POST /users/:id/make_new_api_key.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
HTTP/1.1 200 OK
Date: Mon, 19 Dec 2011 11:45:25 GMT
Server: Apache/2.2.3 (CentOS)
X-Powered-By: Phusion Passenger (mod_rails/mod_rack) 3.0.1
ETag: "781538aa145a713491a20e87b7c6e989"
X-UA-Compatible: IE=Edge,chrome=1
X-Runtime: 0.436403
Cache-Control: max-age=0, private, must-revalidate
Set-Cookie: _session_id=a0f5aa6334b23a1c012bcede1a06a83a7; path=/; HttpOnly
Status: 200
Connection: close
Content-Type: application/xml; charset=utf-8

<user>
  <used_cpus type="integer">0</used_cpus>
  <payment_amount type="decimal">0.0</payment_amount>
  <created_at type="datetime">2011-12-19T11:41:17Z</created_at>
  <activated_at type="datetime">2011-12-19T11:41:17Z</activated_at>
  <used_cpu_shares type="integer">0</used_cpu_shares>
  <suspend_at nil="true"></suspend_at>
  <remember_token_expires_at nil="true"></remember_token_expires_at>
  <cdn_account_status>ACTIVE</cdn_account_status>
  <updated_at type="datetime">2011-12-19T11:45:25Z</updated_at>
  <deleted_at nil="true"></deleted_at>
  <cdn_status>INACTIVE</cdn_status>
  <billing_plan_id type="integer">1</billing_plan_id>
  <api_key>33094u3049rj3409ru409j409r3jr4j309j</api_key>
  <update_billing_stat type="boolean">false</update_billing_stat>
...</

87.29 Search Users by Name

To search a user by name, last name or username, use the following request:
GET http://onapp.test/users.xml?q=name
GET http://onapp.test/users.json?q=name

XML Request Example


JSON Request Example


87.30 Get List of User's YubiKeys

To get the list of Yubikeys assigned to a certain user, use the following request:
GET /users/:user_id/yubi_keys.xml
GET /users/:user_id/yubi_keys.json
XML Request Example

```
curl -i -X GET  -H 'Content-type: application/xml' -u user:userpass  
http://onapp.test/users/984/yubi_keys.xml
```

JSON Request Example

```
curl -i -X GET  -H 'Content-type: application/json' -u user:userpass  
http://onapp.test/users/984/yubi_keys.json
```

XML Output Example

```
<yubi_keys type="array">
    <yubi_key>
        <created_at type="datetime">2016-10-26T14:18:13+00:00</created_at>
        <id type="integer">8</id>
        <label>test.m</label>
        <last_used nil="true"/>
        <otp>nfnonoinnni</otp>
        <updated_at type="datetime">2016-10-26T14:18:13+00:00</updated_at>
        <user_id type="integer">984</user_id>
    </yubi_key>
    <yubi_key>...</yubi_key>
</yubi_keys>
```

Where:

- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **id** - the Yubikey ID
- **label** - the YubiKey name
- **last_used** - the time and date when the YubiKey was last used
- **otp** - the character string generated by the Yubikey
- **updated_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **user_id** - the ID of the user with which this YubiKey is associated

**87.31 Add Yubikey to User**

To add a Yubikey to a user account, use the following request:

POST /users/:user_id/yubi_keys.xml
POST /users/:user_id/yubi_keys.json

XML Request Example

```
curl -i -X POST  -H 'Content-type: application/xml' -u user:userpass  
http://onapp.test/users/12/yubi_keys.xml  -d  
'"<yubi_key><label>1</label><otp>bhbhinoiniubuyvbi</otp></yubi_key>'
```

JSON Request Example
Where:

*label* - the YubiKey name, this parameter is optional

*otp* - the character string generated by the YubiKey

When you enter the Yubikey, the request is sent automatically.

### 87.32 Delete User Yubikey

To delete a Yubikey that is assigned to a certain user, apply the following request:

```
DELETE /users/:user_id/yubi_keys/:yubikey_id.xml
DELETE /users/:user_id/yubi_keys/:yubikey_id.json
```

**XML Request Example**

```
curl -i -X DELETE -H 'Content-type: application/xml' -u user:userpass
http://onapp.test/users/12/yubi_keys/13.xml
```

**JSON Request Example**

```
curl -i -X DELETE -H 'Content-type: application/json' -u user:userpass
http://onapp.test/users/12/yubi_keys/13.json
```

Be careful when deleting a YubiKey as it will no longer be possible to log in using that Yubikey unless you add it again to the system.
88 Users with Config Problems

With OnApp you can manage users which have some configuration problems and resolve those issues through the **Users with config problems** menu.

You can get the following users' lists:

- Users without roles
- Users without timezones
- Users without user groups

88.1 Get the list of users without roles

To view the list of users without roles, use the following request:

GET /users/config_issues.xml
GET /users/config_issues.json

XML Request Example

```
```

JSON Request Example

```
```

As a result, you will get the list of users without roles. For detail on parameters returned, refer to **Get List of Users** section.

88.2 Get the list of users without timezones

To view the list of users without timezones, use the following request:

GET /users/config_issues.xml?issue=without_time_zones
GET /users/config_issues.json?issue=without_time_zones

XML Request Example

```
```

JSON Request Example
As a result, you will get the list of users without timezones. For detail on parameters returned, refer to Get List of Users section.

88.3 Get the list of users without user groups

To view the list of users without user groups, use the following request:

GET /users/config_issues.xml?issue=without_user_groups
GET /users/config_issues.json?issue=without_user_groups

XML Request Example


JSON Request Example


As a result, you will get the list of users without user groups. For detail on parameters returned, refer to Get List of Users section.
89 Version

To check the version of your cloud installation, use the following request:

GET /version.xml
GET /version.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**

```
<onapp>
 <version>2.2</version>
</onapp>
```

**Where:**

*version* - the version of your cloud installation
90 Virtual Servers

Virtual servers in OnApp are based on templates and deployed on compute resources. Virtual servers have their own root accounts, so that VS owners can fully control, configure and manage their servers. All CRUD operations are possible for the virtual servers class.

For details how to manage virtual server custom variables, refer to the Custom Recipe Variables section of this guide.

90.1 Get List of VSs

There are several ways to request the list of VSs. Below you can find the following examples:

- the list of virtual servers with all related resources
- the list of virtual servers

To get the list of virtual servers with all related resources, use the following request:

GET /virtual_machines.xml
GET /virtual_machines.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example
<virtual_machines>
  <virtual_machine>
    <add_to_marketplace nil="true"></add_to_marketplace>
    <admin_note nil="true"></admin_note>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <booted type="boolean">true</booted>
    <built type="boolean">true</built>
    <cores_per_socket type="integer">0</cores_per_socket>
    <cpu_shares type="integer">1</cpu_shares>
    <cpu_threads nil="true"/>
    <cpu_units type="integer">200</cpu_units>
    <created_at type="datetime">2011-11-01T17:11:58+03:00</created_at>
    <customer_network_id nil="true"/>
    <deleted_at nil="true"/>
    <edge_server_type nil="true"/>
    <enable_autoscale type="boolean">true</enable_autoscale>
    <enable_monitis type="boolean">true</enable_monitis>
    <firewall_notrack type="boolean">false</firewall_notrack>
    <fqdn>autobackup</fqdn>
    <hot_add_cpu nil="true"/>
    <hot_add_memory nil="true"/>
    <hypervisor_id type="integer">2</hypervisor_id>
    <id type="integer">000</id>
    <initial_root_password>791791</initial_root_password>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <instance_package_id nil="true"/>
    <iso_id nil="true"/>
    <label>YR_autobackup</label>
    <local_remote_access_ip_address>000.000.000.000</local_remote_access_ip_address>
    <local_remote_access_port type="integer">0000</local_remote_access_port>
    <locked type="boolean">false</locked>
    <memory type="integer">1632</memory>
    <min_disk_size type="integer">5</min_disk_size>
    <note nil="true"></note>
    <operating_system>linux</operating_system>
    <operating_system_distro>rhel</operating_system_distro>
    <preferred_hvs type="array"/>
    <recovery_mode type="boolean">false</recovery_mode>
    <remote_access_password>os3ajolb1buj</remote_access_password>
    <service_password nil="true"/>
    <state>new</state>
    <storage_server_type nil="true"/>
    <strict_virtual_machine_id nil="true"></strict_virtual_machine_id>
    <suspended type="boolean">false</suspended>
    <template_id type="integer">8</template_id>
    <template_label>CentOS 5.6 x86</template_label>
    <template_version>1.5</template_version>
    <time_zone>Atlantic Time (Canada)</time_zone>
    <updated_at type="datetime">2015-03-04T22:06:21+02:00</updated_at>
    <user_id type="integer">5</user_id>
    <vip nil="true"></vip>
    <xen_id type="integer">12</xen_id>
    <virsh_console type="boolean">false</virsh_console>
    <ip_addresses type="array">
      <ip_address>000.000.000.000</ip_address>
    </ip_addresses>
  </virtual_machine>
</virtual_machines>
<gateway>000.000.000.000</gateway>
<hypervisor_id nil="true"/>
{id type="integer">2</id>
<ip_address_pool_id nil="true"/>
<network_address>000.000.000.000</network_address>
<network_id type="integer"/>1</network_id>
<updated_at type="datetime">2011-11-01T17:39:13+03:00</updated_at>
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>000.000.000.000</netmask>
</ip_address>
</ip_addresses>
<monthly_bandwidth_used>0</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>
<price_per_hour type="float">0.0</price_per_hour>
<price_per_hour_powered_off type="float">0.0</price_per_hour_powered_off>
<support_incremental_backups type="boolean">false</support_incremental_backups>
<cpu_priority type="integer">1</cpu_priority>
<built_from_iso type="boolean">true</built_from_iso>
<acceleration type="boolean">false</acceleration>
<hypervisor_type>kvm</hypervisor_type>
</virtual_machine>
...
</virtual_machine>

Where:

- **acceleration** - true if acceleration is enabled for the VS; otherwise false
- **add_to_marketplace** - empty for VSs; used for edge servers only
- **admin_note** - an optional note of the administrator
- **allowed_hot_migrate** - true if the template, on which the VS is based, supports hot migration; otherwise false
- **allowed_swap** - true if swap disk is allowed (depends on the template the VS is based on); otherwise false
- **booted** - true if the VS is running, otherwise false
- **built** - true if the VS is built, otherwise false
- **built_from_iso** - true if the VS is built from ISO; otherwise false
- **cores_per_socket** - the amount of cores per socket
- **cpu_shares** - CPU priority in percent's
- **cpu_sockets** - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted
- **cpu_threads** - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted
- **cpu_units** - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket.
- **cpus** - the number of allocated CPU cores
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **customer_network_id** - ID of a customer network
- **deleted_at** - time when the VS was deleted
**edge_server_type** - true if this is the edge server

**enable_autoscale** - true if autoscaling is allowed for this VS

**enable_monitis** - deprecated attribute; will be removed in upcoming release

**firewall_notrack** - true if the NOTRACK rule is set in iptables

**fqdn** - the name of your host

**hot_add_cpu** - true, if the CPU parameter can be changed without rebooting the VS, otherwise false

**hot_add_memory** - true, if the memory parameter can be changed without rebooting the VS, otherwise false

**hypervisor_id** - the ID of the compute resource used by this VS

**hypervisor_type** - the type of the compute resource the VS is built on (for example: xen, kvm, vcloud, vmware)

**id** - the VS ID

**identifier** - the VS identifier

**initial_root_password** - the VS root password

**initial_root_password_encrypted** - true, if the root password is encrypted, otherwise false.

**instance_package_id** - ID of the instance package

**iso_id** - the ID of the ISO the VS is based on

**label** - the VS label

**local_remote_access_ip_address** - IP address used for remote access

**local_remote_access_port** - the port ID used for console access

**locked** - true if the VS is locked; otherwise false

**memory** - the RAM size allocated to this VS

**min_disk_size** - the minimum disk size required to build a VS from a specified template

**note** - an optional reminder for this VS made by a user account

**operating_system** - operating system used by the VS

**operating_system_distro** - the distribution of the OS from which this VS is built

**preferred_hvs** - the array of preferable compute resources based on compute zone that meet some VS configuration settings

**recovery_mode** - true if recovery mode allowed. Otherwise false

**remote_access_password** - the password for the remote access

**service_password** - service account password

**state** – parameter reserved for future use

**storage_server_type** - true if this is a storage server

**strict_virtual_machine_id** - the ID of a virtual server that will never reside on the same compute resource with this VS

**suspended** - true if VS is suspended, otherwise false

**template_id** - the ID of the template the VS is based on

**template_label** - the name of the template from which this VS is built

**template_version** - the version of the template from which this VS is built

**time_zone** - the time zone set for the VS. This parameter is applicable only to Windows KVM and XEN virtual servers.
Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows VS manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a VS after starting it if time synchronization is not completed for some reason.

`updated_at` - the date when the VS was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format

`user_id` - the ID of a user assigned to this VS

`vip` - true if the VS has VIP status (gives migration priority)

`xen_id` - the VS ID set by the virtualization engine

💡 `virsh_console` - true, if Virsh console is enabled for the VS, otherwise, false

`ip_addresses` - an array of IP addresses assigned to this VS and their details:
  - `address` - IP address
  - `broadcast` - broadcast address
  - `created_at` - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
  - `customer_network_id` - the ID of the customer VLAN the IP address belongs to
  - `disallowed_primary` - true if not allowed to be used as primary, otherwise false
  - `gateway` - gateway address
  - `hypervisor_id` - the ID of a compute resource the IP address is associated with
  - `id` - the ID of the IP address
  - `ip_address_pool_id` - ID of the IP address pool the IP address is associated with
  - `network_address` - the address of the network
  - `network_id` - the ID of the network
  - `pxe` - true, if this address can be used for cloudbooting a compute resource
  - `updated_at` - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format
  - `user_id` - the ID of the user this IP address is assigned to
  - `free` - true if free, otherwise, false
  - `netmask` - netmask for the IP address

`monthly_bandwidth_used` - VS monthly bandwidth in KB

`total_disk_size` - the total disk size in GB of all disks assigned to VS

`price_per_hour` - server's price per hour

`price_per_hour_powered_off` - price per hour when server is powered off

`support_incremental_backups` - 1, if virtual server supports incremental backups, and 0 if it does not

`cpu_priority` - this is a new parameter reserved for further use; currently will have the same value as `cpu_shares`

To get the list of VSs, use the following request:

GET /virtual_machines.xml?short

GET /virtual_machines.json?short
XML Output example

```xml
<virtual_machines type="array">
  <virtual_machine>
    <add_to_marketplace nil="true" />
    <admin_note nil="true" />
    <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <booted type="boolean">false</booted>
    <built type="boolean">false</built>
    <cores_per_socket type="integer">0</cores_per_socket>
    <cpu_shares type="integer">100</cpu_shares>
    <cpu_sockets nil="true" />
    <cpu_threads nil="true" />
    <cpu_units type="integer">1000</cpu_units>
    <cpus type="integer">1</cpus>
    <created_at type="datetime">2015-06-16T19:41:01+00:00</created_at>
    <customer_network_id nil="true" />
    <deleted_at nil="true" />
    <edge_server_type nil="true" />
    <enable_autoscale nil="true" />
    <enable_monitis type="boolean">false</enable_monitis>
    <firewall_notrack type="boolean">false</firewall_notrack>
    <fqdn>example.onappdemo.com</fqdn>
    <hot_add_cpu nil="true" />
    <hot_add_memory nil="true" />
    <hypervisor_id type="integer">1</hypervisor_id>
    <id type="integer">1</id>
    <identifier>s34b4zqkh3gki</identifier>
    <initial_root_password>ycW50dZ0ryjj</initial_root_password>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <instance_package_id nil="true" />
    <iso_id nil="true" />
    <label>example.onappdemo.com</label>
    <local_remote_access_ip_address nil="true" />
    <local_remote_access_port nil="true" />
    <locked type="boolean">false</locked>
    <memory type="integer">512</memory>
    <min_disk_size type="integer">5</min_disk_size>
    <note nil="true" />
    <operating_system>linux</operating_system>
    <operating_system_distro>centos</operating_system_distro>
    <preferred_hvs type="array" />
    <recovery_mode nil="true" />
    <remote_access_password nil="true" />
    <service_password nil="true" />
    <state>new</state>
    <storage_server_type nil="true" />
    <strict_virtual_machine_id nil="true" />
    <suspended type="boolean">false</suspended>
    <template_id type="integer">4</template_id>
    <template_label>CentOS 5.3</template_label>
    <template_version>1.5</template_version>
    <time_zone nil="true" />
    <updated_at type="datetime">2015-06-16T19:41:02+00:00</updated_at>
    <user_id type="integer">3</user_id>
    <vip nil="true" />
    <xen_id nil="true" />
  </virtual_machine>
</virtual_machines>
```
v.6.1 Edge 2
- added the *virsh_console* parameter

v.6.0
- removed the *acceleration_status* parameter

v.5.8
- replaced the *hostname* parameter with *fqdn* parameter

v.5.3
- added the *template_version* parameter

v.4.3
- added the *hypervisor_type* parameter

v.4.2
- added the following parameters:
  - *acceleration*
  - *acceleration_status*
  - *built_from_iso*

v.4.1
- added the *time_zone* parameter

### 90.2 Get VS Details

To get the details of a particular virtual server, use the following request:

GET /virtual_machines/:id.xml
GET /virtual_machines/:id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**XML Output Example**
<virtual_machine>
    <add_to_marketplace nil="true"></add_to_marketplace>
    <admin_note nil="true"></admin_note>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <booted type="boolean">true</booted>
    <built type="boolean">true</built>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <allowed_swap type="boolean">true</allowed_swap>
    <booted type="boolean">true</booted>
    <built type="boolean">true</built>
    <cores_per_socket type="integer">0</cores_per_socket>
    <cpu_shares type="integer">1</cpu_shares>
    <cpu_sockets nil="true"></cpu_sockets>
    <cpu_threads nil="true"></cpu_threads>
    <cpu_units type="integer">200</cpu_units>
    <cpus type="integer">1</cpus>
    <created_at type="datetime">2011-11-01T17:11:58+03:00</created_at>
    <customer_network_id nil="true"></customer_network_id>
    <deleted_at nil="true"></deleted_at>
    <domain>test.com</domain>
    <edge_server_type nil="true"></edge_server_type>
    <enable_autoscale type="boolean">true</enable_autoscale>
    <enable_monitis type="boolean">true</enable_monitis>
    <firewall_notrack type="boolean">false</firewall_notrack>
    <fqdn>autobackup</fqdn>
    <hot_add_cpu nil="true"></hot_add_cpu>
    <hot_add_memory nil="true"></hot_add_memory>
    <hypervisor_id type="integer">2</hypervisor_id>
    <id type="integer">100</id>
    <identifier>iskngs9dve0hdg</identifier>
    <initial_root_password>791791</initial_root_password>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <instance_package_id nil="true"></instance_package_id>
    <iso_id nil="true"></iso_id>
    <label>YR_autobackup</label>
    <local_remote_access_ip_address>000.000.000.000</local_remote_access_ip_address>
    <local_remote_access_port type="integer">0000</local_remote_access_port>
    <locked type="boolean">false</locked>
    <memory type="integer">1632</memory>
    <min_disk_size type="integer">5</min_disk_size>
    <note nil="true"></note>
    <operating_system>linux</operating_system>
    <operating_system_distro>rhel</operating_system_distro>
    <preferred_hvs type="array"/>
    <recovery_mode type="boolean">false</recovery_mode>
    <remote_access_password>os3ajolb1buj</remote_access_password>
    <service_password nil="true"></service_password>
    <state>new</state>
    <storage_server_type nil="true"></storage_server_type>
    <strict_virtual_machine_id nil="true"></strict_virtual_machine_id>
    <suspended type="boolean">false</suspended>
    <template_id type="integer">8</template_id>
    <template_label>CentOS 5.6 x86</template_label>
    <template_version>1.5</template_version>
    <time_zone>Atlantic Time (Canada)</time_zone>
    <updated_at type="datetime">2015-03-04T12:06:21+02:00</updated_at>
    <user_id type="integer">5</user_id>
    <vip nil="true"></vip>
    <xen_id type="integer">12</xen_id>
    <virsh_console type="boolean">false</virsh_console>
    <ip_addresses type="array"/>
    <ip_address type="array"/>
    <address>000.000.000.000</address>
    <broadcast>000.000.000.000</broadcast>
    <created_at type="datetime">2011-10-10T12:31:12+03:00</created_at>
    <customer_network_id nil="true"></customer_network_id>
    <disallowed_primary type="boolean">false</disallowed_primary>
<gateway>000.000.000.000</gateway>
<hypervisor_id nil="true"/>
{id type="integer">2</id>
<ip_address_pool_id nil="true"/>
<network_address>000.000.000.000</network_address>
<network_id type="integer">1</network_id>
<pxe type="boolean">false</pxe>
<updated_at type="datetime">2011-11-01T17:39:13+03:00</updated_at>
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>000.000.000.000</netmask>
</ip_address>
</ip_addresses>
</monthly_bandwidth_used>0</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>
<price_per_hour type="float">0.0</price_per_hour>
<price_per_hour_powered_off type="float">0.0</price_per_hour_powered_off>
<support_incremental_backups type="boolean">false</support_incremental_backups>
<cpu_priority type="integer">1</cpu_priority>
<built_from_iso type="boolean">true</built_from_iso>
<acceleration_allowed type="boolean">false</acceleration_allowed>
<hypervisor_type>kvm</hypervisor_type>
</virtual_machine>

**Where:**

- `acceleration_allowed` - true if acceleration is enabled for the VS; otherwise false
- `add_to_marketplace` - empty for VSs; used for edge servers only
- `admin_note` - an optional note of the administrator
- `allowed_hot_migrate` - true if the template, on which the VS is based, supports hot migration; otherwise false
- `allowed_swap` - true if swap disk is allowed (depends on the template the VS is based on); otherwise false
- `booted` - true if the VS is running, otherwise false
- `built` - true if the VS is built, otherwise false
- `built_from_iso` - true if the VS is built from ISO; otherwise false
- `cores_per_socket` - the amount of cores per socket
- `cpu_shares` - CPU priority in percent's
- `cpu_sockets` - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted
- `cpu_threads` - the amount of CPU threads per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted
- `cpu_units` - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket.
- `cpus` - the number of allocated CPU cores
- `created_at` - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- `customer_network_id` - theID of a customer network
- `deleted_at` - time when the VS was deleted
- `domain` - the domain for this VS. The default value is `localdomain`. This parameter is not applicable for Windows virtual servers.
edge_server_type - true if this is the edge server
enable_autoscale - true if autoscaling is allowed for this VS
enable_monitis - deprecated attribute; will be removed in upcoming release
firewall_notrack - true if the NOTRACK rule is set in iptables
fqdn - the name of your host
hot_add_cpu - true, if the CPU parameter can be changed without rebooting the VS, otherwise, false
hot_add_memory - true, if the memory parameter can be changed without rebooting the VS, otherwise false
hypervisor_id - the ID of the compute resource used by this VS
hypervisor_type - the type of the compute resource the VS is built on (for example: xen, kvm, vcloud, vmware)
id - the VS ID
identifier - the VS identifier
initial_root_password - the VS root password
initial_root_password_encrypted - true, if the root password is encrypted, otherwise, false
instance_package_id - ID of the instance package
iso_id - the ID of the ISO the VS is based on
label - the VS label
local_remote_access_ip_address - IP address used for remote access
local_remote_access_port - the port ID used for console access
locked - true if the VS is locked; otherwise false
memory - the RAM size allocated to this VS
min_disk_size - the minimum disk size required to build a VS from a specified template
note - an optional reminder for this VS made by a user account
operating_system - operating system used by the VS
operating_system_distro - the distribution of the OS from which this VS is built
preferred_hvs - the array of preferable compute resources based on compute zones zone that meet some VS configuration settings
recovery_mode - true if recovery mode allowed. Otherwise false
remote_access_password - the password for the remote access
service_password - service account password
state – parameter reserved for future use
storage_server_type - true if this is a storage server
strict_virtual_machine_id - the ID of a virtual server that will never reside on the same compute resource with this VS
suspended - true if VS is suspended, otherwise false
template_id - the ID of the template the VS is based on
template_label - the name of the template from which this VS is built
template_version - the version of the template from which this VS is built
time_zone - the time zone set for the VS. This parameter is applicable only to Windows KVM and XEN virtual servers.
Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows VS manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a VS after starting it if time synchronization is not completed for some reason.

- **updated_at** - the date when the VS was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **user_id** - the ID of a user assigned to this VS
- **vip** - true if the VS has VIP status (gives migration priority)
- **xen_id** - the VS ID set by the virtualization engine
- **virsh_console** - true, if Virsh console is enabled for the VS, otherwise, false
- **ip_addresses** - an array of IP addresses assigned to this VS and their details:
  - **address** - IP address
  - **broadcast** - broadcast address
  - **created_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - **customer_network_id** - the ID of the customer VLAN the IP address belongs to
  - **disallowed_primary** - true if not allowed to be used as primary, otherwise false
  - **gateway** - gateway address
  - **hypervisor_id** - the ID of a compute resource the IP address is associated with
  - **id** - the ID of the IP address
  - **ip_address_pool_id** - ID of the IP address pool the IP address is associated with
  - **network_address** - the address of the network
  - **network_id** - the ID of the network
  - **pxe** - true, if this address can be used for cloudbooting a compute resource
  - **updated_at** - the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
  - **user_id** - the ID of the user this IP address is assigned to
  - **free** - true if free, otherwise false
  - **netmask** - netmask for the IP address
- **monthly_bandwidth_used** - VS monthly bandwidth in Gb
- **total_disk_size** - the total disk size in GB of all disks assigned to VS
- **price_per_hour** - server's price per hour
- **price_per_hour_powered_off** - price per hour when server is powered off
- **support_incremental_backups** - 1, if virtual server supports incremental backups, and 0 if it does not
- **cpu_priority** - this is a new parameter reserved for further use; currently will have the same value as cpu_shares

**Page History**

v.6.1 Edge 2
- added the virsh_console parameter
v.6.1 Edge 1
- removed the `acceleration_status` parameter

v.6.0
- replaced the `acceleration` parameter with `acceleration_allowed` parameter

v.5.8
- replaced the `hostname` parameter with `fqdn` parameter

v.5.3
- added `template_version` parameter

v.4.3
- added `hypervisor_type` parameter

v.4.2
- added the following parameters:
  - added `acceleration` parameter
  - added `acceleration_status` parameter
  - added `built_from_iso` parameter

v.4.1
- added `time_zone` parameter

90.3 Get Statuses for all Virtual Servers

To get statuses for all virtual servers, use the following request:

GET /virtual_machines/status.xml
GET /virtual_machines/status.json

XML Request example

```
```

JSON Request example

```
```

XML Output example
Where:

- **id** - virtual server ID
- **identifier** — the VS identifier
- **hostname** — the name of your host
- **template_id** — the ID of the template the VS is based on
- **built** — true if the VS is built, otherwise false
- **locked** — true if the VS is locked; otherwise false
- **booted** — true if the VS is running, otherwise false
- **operating_system** — operating system used by the VS
- **suspended** — true if VS is suspended, otherwise false
- **enable_autoscale** — true if autoscaling is allowed for this VS
- **state** — virtual server state

### 90.4 Get Virtual Server Status

This parameter has been added in the 3.1 version.

To get status for a particular virtual server, use the following request:

GET /virtual_machines/:virtual_machine_id/status.xml
GET /virtual_machines/:virtual_machine_id/status.json

**XML Request Example**

```
```

**JSON Request Example**

XML Output Example

```xml
<virtual_machine>
  <id type="integer">48</id>
  <identifier>b266b5h5et39iy</identifier>
  <hostname>qaaoxp</hostname>
  <template_id type="integer">111</template_id>
  <built type="boolean">true</built>
  <locked type="boolean">false</locked>
  <booted type="boolean">true</booted>
  <operating_system>windows</operating_system>
  <suspended type="boolean">false</suspended>
  <enable_autoscale type="boolean">false</enable_autoscale>
  <state>new</state>
</virtual_machine>
```

Where:

- `id` - virtual server ID
- `identifier` — the VS identifier
- `hostname` — the name of your host
- `template_id` — the ID of the template the VS is based on
- `built` — true if the VS is built, otherwise false
- `locked` — true if the VS is locked; otherwise false
- `booted` — true if the VS is running, otherwise false
- `operating_system` — operating system used by the VS
- `suspended` — true if VS is suspended, otherwise false
- `enable_autoscale` — true if autoscaling is allowed for this VS
- `state` — virtual server state

### 90.5 Get VS Acceleration Status

To get current acceleration status for a virtual server, use the following request:

**GET /virtual_machines/:id/acceleration/status.xml**

**GET /virtual_machines/:id/acceleration/status.json**

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```
curl i -X GET -u user:userpass --url
http://onapp.test/virtual_machines/xyvbnejmoypiah/acceleration/status.json
-H 'Accept: application/json' -H 'Content-type: application/json'

XML Output Example

<?xml version="1.0" encoding="UTF-8"?>
<hash>
  <acceleration_status>Active</acceleration_status>
</hash>

Where:

acceleration_status - the status of acceleration: active or inactive

90.6 Add VS

To add a VS, use the following request:

POST /virtual_machines.xml
POST /virtual_machines.json

This section describes the API request that adds a VS and sets its resources. For information on the API request that adds a VS using instance packages, refer to [Add Instance Package VS](#).

XML Request example
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
'<virtual_machine><template_id>8</template_id><label>zaza</label><hostname>zaza</hostname><domain>localdomain</domain><hypervisor_group_id>14</hypervisor_group_id><hypervisor_id>1</hypervisor_id><initial_root_password>ehgebhewtwh</initial_root_password><memory>128</memory><cpu>1</cpu><cpu_shares>1</cpu_shares><cpusockets>12</cpusockets><datastoregroupprimaryid>18</datastoregroupprimaryid><primarydisksize>5</primarydisksize><primarydiskminior>100</primarydiskminior><locationid>12</locationid><licensing_server_id>38</licensing_server_id><licensing_type>kms</licensing_type><licensing_key>keyexample</licensing_key><datastoregroupswapid>18</datastoregroupswapid><swappid>1</swappid><swapdisksize>1</swapdisksize><swapdiskminior>100</swapdiskminior><primarynetworkgroupid>19</primarynetworkgroupid><selectedipaddress>5.1.1.12</selectedipaddress><ratenlimit>1</ratenlimit><requiredipaddressassignment>1</requiredipaddressassignment><requiredautomaticbackup>0</requiredautomaticbackup><requiredvirtualmachinebuild>1</requiredvirtualmachinebuild><requiredvirtualmachinesetup>1</requiredvirtualmachinesetup><timezone>Atlantic Time (Canada)</timezone><enableautoscale>0</enableautoscale><accelerationallowed>true</accelerationallowed><customrecipevariables><customrecipevariable><name>varname</name><value>value</value><enabled>1</enabled></customrecipevariable></customrecipevariables><serviceaddonids type="array"><serviceaddonid>273</serviceaddonid><serviceaddonid>274</serviceaddonid></serviceaddonids>'
--url http://onapp.test/virtual_machines.xml

**JSON Request example**

```json

```

---

1064
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '

{"virtual_machine": {
  "acceleration_allowed": "true",
  "cpu_shares": "1",
  "cpu_sockets": "12",
  "cpus": "1",
  "custom_recipe_variables_attributes": [
    {
      "enabled": "1",
      "name": "var_name_1",
      "value": "var_value_1"
    },
    {
      "enabled": "1",
      "name": "var_name_2",
      "value": "var_value_2"
    }
  ],
  "service_addon_ids": [
    273,
    274
  ],
  "data_store_group_primary_id": "18",
  "data_store_group_swap_id": "18",
  "domain": "localdomain",
  "enable_autoscale": "0",
  "hostname": "zaza",
  "hypervisor_group_id": "14",
  "hypervisor_id": "1",
  "initial_root_password": "tyrhshghj657th",
  "label": "zaza",
  "licensing_key": "keyexample",
  "licensing_server_id": "38",
  "licensing_type": "kms",
  "location_id": "12",
  "memory": "128",
  "primary_disk_min_iops": "100",
  "primary_disk_size": "5",
  "primary_network_group_id": "19",
  "rate_limit": "1",
  "recipe_joins_attributes": {
    "123": {"recipe_id": "11"}
  },
  "required_automatic_backup": "0",
  "required_ip_address_assignment": "1",
  "required_virtual_machine_build": "1",
  "required_virtual_machine_startup": "1",
  "selected_ip_address": "5.1.1.12",
  "swap_disk_min_iops": "100",
  "swap_disk_size": "1",
  "template_id": "8",
  "time_zone": "Atlantic Time (Canada)"
}}' --url http://onapp.test/virtual_machines.json

Where:

acceleration_allowed - true if acceleration is enabled for the VS; otherwise false

memory* - amount of RAM assigned to the VS, MB
cpus* - number of CPUs assigned to the VS. For KVM compute resources, this parameter sets CPU sockets by default, unless CPU topology is enabled.

cpu_shares* - required parameter. For KVM compute resource the CPU priority value is always 100. For XEN, set a custom value. The default value for XEN is 1

cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket.

cpu_sockets - the amount of CPU sockets. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted.

hostname* - set the host name for this VS

domain - specify the domain for this VS. The default value is localdomain. You can edit the default value for domain in /onapp/interface/config/on_app.yml. This parameter is not applicable to Windows virtual servers.

instance_package_id - ID of the instance package that will be used to build the VS

label* - user-friendly VS description

location_id - set the ID of the location where the VS should be created

primary_disk_size* - set the disk space for this VS

swap_disk_size* - set swap space. There is no swap disk for Windows-based VSs

primary_disk_min_iops - minimum number of IO operations per second for primary disk (this is a SolidFire related parameter)

swap_disk_min_iops - minimum number of IO operations per second for swap disk (this is a SolidFire related parameter)

type_of_format - type of filesystem - ext4. For Linux templates, you can choose ext4 file system instead of the ext3 default one

data_store_group_primary_id - set the ID of the data store zone to which this primary disk is allocated

data_store_group_swap_id - set the ID of the data store zone to which this swap disk is allocated

network_id - the ID of the primary network. Optional parameter that can be used only if it is assigned to the network zone.

primary_network_group_id - the ID of the primary network group. Optional parameter

required_automatic_backup - set 1 if you need automatic backups

rate_limit - set max port speed in Mbps or set 0 to get maximum port speed allowed by your bucket. If this parameter is omitted or sent without value, the default port speed will be configured for the VS. The default port speed depends on the maximum port speed set in your bucket and the Max network interface port speed parameter at Control Panel > Settings > Configuration. The system identifies which of the two values (in the bucket or in the configuration) is lower and sets it as the default port speed during VS creation.

required_virtual_machine_build * - set 1 to build VS automatically

required_virtual_machine_startup - set 1 to start up the VS automatically, otherwise set 0 (default state is "1")

time_zone - the time zone set for the VS. This parameter is applicable only to Windows virtual servers.

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows VS manually. Setting correct time zone at the Compute resource side helps to
keep correct time inside a VS after starting it if time synchronization is not completed for some reason.

*required_ip_address_assignment* - set "1" if you want VS to be created with already assigned IP address, otherwise set "0"

If you set the *required_ip_address_assignment* parameter to "0", you need to indicate an IP address for the server in the *selected_ip_address* parameter, otherwise, the VS will not be created.

*selected_ip_address* - an IP address to assign to this VS

*admin_note* - enter a brief comment for the VS. Optional parameter

*note* - a brief comment a user can add to a VS

*template_id* - the ID of a template from which a VS should be built

Make sure that a template is located on a backup server attached to the compute resource on which you wish to built the VS, otherwise, the creation of the VS will fail.

*licensing_server_id* - the ID of a template group where the KMS server details are indicated and to which the template belongs (either directly or through the child group). This parameter is required for Windows virtual machines with KMS licensing type only

*licensing_type* - the type of a license: mak, kms or user own license. This parameter is required for Windows virtual machines only

*licensing_key* - the key of a license, required if you have selected own licensing type, and not required for MAK and KMS licensing types

*hypervisor_group_id* - the ID of the compute zone in which the VS will be created. Optional: if no compute zone is set, the VS will be built in any available compute zone

*hypervisor_id* - the ID of a compute resource where the VS will be built. Optional: if no compute resource ID is specified, the VS will be built on the compute resource with the least available RAM (but sufficient RAM for the VS)

*initial_root_password* - the root password for a VS. Optional, if none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [__], and the following special characters: ~ ! @ # $ * _ + = " { } [ ] : ; ' , . ? / . You can use both lower- and uppercase letters.

The following characters are not allowed for Windows-based virtual servers:

- percent sign [%]
- double quotation marks ['"]
- brackets [,<,>]
- vertical bar [|
- caret [^]
- ampersand [&]
• parentheses [(),]

initial_root_password_encryption_key - specify the password encryption passphrase

recipe_joins_attributes - an array of recipes to run on the virtual server provisioning with the details:

• sequence_number - must be unique
  o recipe_id - the ID of the recipe

custom_recipe_variables_attributes - an array of custom variables with the details.

• enabled - true, if the variable is enabled, otherwise false
• name - variable name
• value - variable value script

service_addon_ids - an array of service add-on IDs, which you want to add to VS

XML Output Example
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>255.255.255.192</netmask>
</ip_address>
</ip_addresses>
<monthly_bandwidth_used>0</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>
</virtual_machine>

Where:

add_to_marketplace - true, if the edge server is added to marketplace. The default value is "false". This parameter is for CDN servers only.

admin_note - administrator comment for the VS

allowed_resize_without_reboot - true if resize without reboot is allowed, otherwise false

allowed_hot_migrate - true if hot migration is allowed, otherwise false

allowed_swap - true if swap is allowed, otherwise false

booted - true, if the VS is booted, otherwise false

built - true, if the VS is built, otherwise false

cpu_shares - the percentage of allocated CPU priority resource

cpus - number of CPUs assigned to the VS

created_at - the date when the VS was created in the [YYYY][MM][DD]T[hh][mm][ss] format

updated_at - the date when the VS was updated in the [YYYY][MM][DD]T[hh][mm][ss] format

customer_network_id - ID of the customer network

edge_server_type - type of the CDN edge server. This parameter is for CDN servers only.

enable_autoscale - true if autoscaling is allowed for this VS

enable_monitis - deprecated attribute

firewall notrack - parameter for adding firewall rules. It true for edge servers only.

hostname - VS hostname

hypervisor_id - the ID of the compute resource, on which the server is deployed

id - the VS ID in OnApp CP database

identifier - the VS identifier

initial_root_password - the VS root password

initial_root_password_encrypted - true, if the root password is encrypted, otherwise false

label - user-friendly VS description

local_remote_access_ip_address - IP address for remote connection

local_remote_access_port - port for remote connection

locked - true if the VS is locked; otherwise false

memory - the RAM size allocated to this VS, MB

min_disk_size - the minimum disk size required to build a VS from a specified template

operating_system - operating system used by the VS

operating_system_distro - the distribution of the OS from which this VS is built
recovery_mode - true if recovery mode allowed, otherwise false
remote_access_password - the password for the remote access
service_password - password of a service user
strict_virtual_machine_id - the ID of a virtual machine that will never reside on the same compute resource with this VS
suspended - true if VS is suspended, otherwise false
template_id - the ID of the template the VS is based on
template_label - the name of the template from which this VS is built
user_id - the ID of a user assigned to this VS
vip - true if the VS has VIP status (gives migration priority)
xen_id - the VS ID set by the virtualization engine
ip_addresses - an array of IP addresses with the following parameters:
  • address - IP address
  • broadcast - broadcast address
  • created_at - the date when the IP address was created in the [Y][YY][M][M][DD][T][hh][mm][ss]Z format
  • disallowed_primary - true if not allowed to be used as primary (for VS build), otherwise false
  • gateway - gateway address
  • id - the ID of the IP address
  • ip_address_pool_id - ID of the IP address pool to the IP address belongs to
  • network_address - the address of the network
  • network_id - the ID of the network
  • updated_at - the date when the IP address was updated in the [Y][YY][M][M][DD][T][hh][mm][ss]Z format
  • user_id - the ID of a user associated with this IP address
  • free - true if free, otherwise false
  • netmask - netmask for the IP address
monthly_bandwidth_used - VS monthly bandwidth in KB
⚠️ virsh_console - set true to enable Virsh console for the VS, otherwise, false
total_disk_size - total VS disk size

Page History
v. 6.1 Edge 2
• added the virsh_console parameter
v.6.0
• replaced location_group_id with location_id parameter
v.5.8
• removed cpu_threads parameter
v.5.5
• added service_addon_ids parameter
v. 5.4
- added the following parameters:
  - domain
  - selected_ip_address
- removed selected_ip_address_id parameter

v. 5.3
- recipe_ids replaced with recipe_joins_attributes parameter
- custom_variables replaced with custom_recipe_variables_attributes parameter

v. 5.2
- recipe_ids will be replaced with recipe_joins_attributes parameter in OnApp 5.3
- custom_variables will be replaced with custom_recipe_variables_attributes parameter in OnApp 5.3

v. 4.1
- added the following parameters:
  - location_group_id
  - time_zone

v. 4.0
- added location_group_id parameter

v. 3.3.2:
- added the following parameters:
  - cpu_sockets
  - cpu_threads

v. 3.3:
- added cpu_units parameter

v. 3.1:
- added the following parameters:
  - custom_variables
  - enabled
  - id
  - name
  - value

90.7 Add VS from OVA Template

To add a VS with multiple disks based on OVA template, use the following request:

POST /virtual_machines.xml
POST /virtual_machines.json

XML Request Example
curl -i -X POST -u user:userpass --url http://onapp.test/virtual_machines.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<virtual_machine><template_id>81</template_id><label>zaza</label><hostname>zaza</hostname><domain>localdomain</domain><initial_root_password>qwaszx!Q2</initial_root_password><initial_root_password_confirmation>qwaszx!Q2</initial_root_password_confirmation><hypervisor_group_id>8</hypervisor_group_id><hypervisor_id>8</hypervisor_id><memory>1024</memory><cpus>1</cpus><cpu_shares>1</cpu_shares><cpu_units>0</cpu_units><cpu_topology>0</cpu_topology><disks_attribute type="array"><disks_attribute><data_store_id>3</data_store_id></disks_attribute><disks_attribute><data_store_id>1</data_store_id></disks_attribute><disks_attribute><data_store_id>1</data_store_id></disks_attribute></disks_attribute><network_interfaces_attribute type="array"><network_interfaces_attribute><network_id>3</network_id><ip_net_id>2</ip_net_id><ip_range_id>2</ip_range_id><ip_address>10.83.37.111</ip_address><rate_limit>1</rate_limit></network_interfaces_attribute><network_interfaces_attribute><network_id>2</network_id><ip_net_id>10.86.37.112</ip_net_id><ip_range_id>1</ip_range_id><ip_address>10.86.37.112</ip_address><rate_limit>1</rate_limit></network_interfaces_attribute><network_interfaces_attribute><network_id>2</network_id><ip_net_id>1</ip_net_id><ip_range_id>0</ip_range_id><ip_address>10.86.37.112</ip_address><rate_limit>0</rate_limit></network_interfaces_attribute><required_automatic_backup>0</required_automatic_backup><required_virtual_machine_build>1</required_virtual_machine_build><required_virtual_machine_startup>1</required_virtual_machine_startup><acceleration>0</acceleration></virtual_machine>'

JSON Request Example

```
curl -i -X POST -u user:userpass --url http://onapp.test/virtual_machines.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"virtual_machine": {"template_id": "81", "label": "zaza", "hostname": "zaza", "domain": "localdomain", "initial_root_password": "qwaszx!Q2", "initial_root_password_confirmation": "qwaszx!Q2", "hypervisor_group_id": "8", "hypervisor_id": "8", "memory": "1024", "cpus": "1", "cpu_shares": "1", "cpu_units": "0", "cpu_topology": "0", "disks_attributes": [{"data_store_id": "3"}, {"data_store_id": "1"}], "network_interfaces_attributes": [{"network_id": "3", "ip_net_id": "2", "ip_range_id": "2", "ip_address": "10.83.37.198", "rate_limit": "1"}, {"network_id": "2", "ip_net_id": "1", "ip_range_id": "1", "ip_address": "10.86.37.199", "rate_limit": "1"}], "required_automatic_backup": "0", "required_virtual_machine_build": "1", "required_virtual_machine_startup": "1", "acceleration": "0"}}'
```

Where:

**template_id** - the ID of a template from which a VS should be built

Make sure that a template is located on a backup server attached to the compute resource on which you wish to build the VS, otherwise, the creation of the VS will fail.

**label** - user-friendly VS description

**hostname** - set the host name for this VS
**domain** - specify the domain for this VS. The default value is localdomain. You can edit the default value for domain in /onapp/interface/config/on_app.yml. This parameter is not applicable for Windows virtual servers.

**initial_root_password** - the root password for a VS. Optional, if none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _], and the following special characters: ~ ! @ # $ * _ + = ` \ | { } : ; ' , . ? / . You can use both lower- and uppercase letters.

The following characters are not allowed for Windows-based virtual servers:

- percent sign [%]
- double quotation marks ['"]
- brackets [<,>]
- vertical bar [||]
- caret [^]
- ampersand [&]
- parentheses [(,)]

**initial_root_password_confirmation** - confirm the root password for a VS

**hypervisor_group_id** - the ID of the compute zone in which the VS will be created. Optional: if no compute zone is set, the VS will be built in any available compute zone

**hypervisor_id** - the ID of a compute resource where the VS will be built. Optional: if no compute resource ID is specified, the VS will be built on the compute resource with the least available RAM (but sufficient RAM for the VS)

**memory** - amount of RAM assigned to the VS

**cpus** - number of CPUs assigned to the VS. For KVM compute resources, this parameter sets CPU sockets by default, unless CPU topology is enabled.

**cpu_shares** - required parameter. For KVM compute resource the CPU priority value is always 100. For XEN, set a custom value. The default value for XEN is 1

**cpu_units** - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket.

**cpu_topology** - set 1 to enable CPU topology, otherwise set 0

**disks_attributes** - an array of disk attributes with the following parameters:

- **data_store_id** - the ID of the data store for VS's disk

**network_interfaces_attributes** - an array of network interfaces attributes with the following parameters:

- **network_id** - the ID of the network from which the VS should get the IP address
- **ip_net_id** - the ID of the IP net from which the IP address should be assigned
- **ip_range_id** - the ID of the IP range from which the IP address should be assigned
- **ip_address** - the ID of an IP address for the VS

Be aware, that you should choose only public IP address. Otherwise, VS built from OVA will not work properly.

**rate_limit** - set max port speed in Mbps or set 0 to get maximum port speed allowed by your bucket. If this parameter is omitted or sent without value, the default port speed will be configured for the VS. The default port speed depends on the maximum port speed set in your bucket and the Max network interface port speed parameter at Control
Panel > Settings > Configuration. The system identifies which of the two values (in the bucket or in the configuration) is lower and sets it as the default port speed during VS creation.

- `required_automatic_backup` - set 1 if you need automatic backups
- `required_virtual_machine_build*` - set 1 to build VS automatically
- `required_virtual_machine_startup` - set 1 to start up the VS automatically, otherwise set 0 (default state is "1")
- `acceleration` - true if acceleration is enabled for the virtual server; otherwise false.

XML Output Example
<virtual_machine>
  <id type="integer">320</id>
  <hypervisor_id type="integer">8</hypervisor_id>
  <template_id type="integer">81</template_id>
  <identifier>atqtomftqjna</identifier>
  <hostname>zaza</hostname>
  <memory type="integer">1024</memory>
  <cpus type="integer">1</cpus>
  <cpu_shares type="integer">1</cpu_shares>
  <created_at type="dateTime">2018-08-28T15:39:16+03:00</created_at>
  <updated_at type="dateTime">2018-08-28T15:39:17+03:00</updated_at>
  <built type="boolean">false</built>
  <locked type="boolean">true</locked>
  <booted type="boolean">false</booted>
  <remote_access_password>1E5nZ0g00P05</remote_access_password>
  <local_remote_access_port nil="true"/>
  <operating_system>linux</operating_system>
  <operating_system_distro>rhel</operating_system_distro>
  <allowed_swap type="boolean">true</allowed_swap>
  <template_label>CentOS Multi disks and networks KVM</template_label>
  <min_disk_size type="integer">1</min_disk_size>
  <allowed_hot_migrate type="boolean">false</allowed_hot_migrate>
  <note nil="true"/>
  <admin_note nil="true"/>
  <suspended type="boolean">false</suspended>
  <strict_virtual_machine_id nil="true"/>
  <enable_autoscale type="boolean">false</enable_autoscale>
  <add_to_marketplace nil="true"/>
  <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
  <storage_server_type nil="true"/>
  <firewall_notrack type="boolean">false</firewall_notrack>
  <service_password nil="true"/>
  <preferred_hvs type="array"/>
  <local_remote_access_ip_address>10.80.0.101</local_remote_access_ip_address>
  <cpu_units type="integer">10</cpu_units>
  <cpu_sockets nil="true"/>
  <draas_keys type="array"/>
  <iso_id nil="true"/>
  <cores_per_socket type="integer">0</cores_per_socket>
  <instance_package_id nil="true"/>
  <hot_add_cpu nil="true"/>
  <hot_add_memory nil="true"/>
  <time_zone nil="true"/>
  <autoscale_service nil="true"/>
  <cdboot type="boolean">false</cdboot>
  <draas_mode type="integer">0</draas_mode>
  <vapp_id nil="true"/>
  <vmware_tools nil="true"/>
  <vcenter_moref nil="true"/>
  <template_version>1.0</template_version>
  <openstack_id nil="true"/>
  <domain>localdomain</domain>
  <vcenter_reserved_memory type="integer">0</vcenter_reserved_memory>
  <deleted_at nil="true"/>
</virtual_machine>
Where:

- **id** - the VS id
- **hypervisor_id** - the ID of a compute resource where the VS will be built. Optional: if no compute resource ID is specified, the VS will be built on the compute resource with the least available RAM (but sufficient RAM for the VS)
**template_id** - the ID of the template the VS is based on

**identifier** - the VS identifier

**hostname** - the host name for this VS

**memory** - amount of RAM assigned to the VS

**cpus** - number of CPUs assigned to the VS. For KVM compute resources, this parameter sets CPU sockets by default, unless CPU topology is enabled.

**cpu_shares** - required parameter. For KVM compute resource the CPU priority value is always 100. For XEN, set a custom value. The default value for XEN is 1

**created_at** - the date in the [YYYY] [MM] [DD][T][hh][mm][ss]Z format

**updated_at** - the date when the VS was updated in the [YYYY] [MM] [DD][T][hh][mm][ss]Z format

**built** - true if the VS is built, otherwise false

**locked** - true if the VS is locked; otherwise false

**booted** - true if the VS is running, otherwise false

**xen_id** - the VS ID set by the virtualization engine

**remote_access_password** - the password for the remote access

**local_remote_access_port** - the port ID used for console access

**label** - the VS label

**recovery_mode** - true if recovery mode allowed. Otherwise false

**user_id** - the ID of a user assigned to this VS

**operating_system** - operating system used by the VS

**operating_system_distro** - the distribution of the OS from which this virtual server is built

**allowed_swap** - true if swap disk is allowed (depends on the template the virtual server is based on); otherwise false

**template_label** - the name of the template from which this virtual server is built

**min_disk_size** - the minimum disk size required to build a VS from a specified template

**allowed_hot_migrate** - true if the template, on which the VS is based, supports hot migration; otherwise false

**note** - an optional reminder for this virtual server made by a user account

**admin_note** - an optional note of the administrator

**suspended** - true if VS is suspended, otherwise false

**strict_virtual_machine_id** - the ID of a virtual server that will never reside on the same compute resource with this VS

**enable_autoscale** - true if autoscaling is allowed for this VS

**add_to_marketplace** - empty for VSs; used for edge servers only

**state** - parameter reserved for future use

**initial_root_password_encrypted** - true, if the root password is encrypted, otherwise false.

**edge_server_type** - true if this is the edge server

**storage_server_type** - true if this is a storage server

**firewall_notrack** - true if the NOTRACK rule is set in iptables

**service_password** - service account password

**preferred_hvs** - the array of preferable compute resources based on compute zone that meet some VS configuration settings
local_remote_access_ip_address - IP address used for remote access

cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in user billing plan.

cpu.Sockets - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted.

iso_id - the ID of the ISO the VS is based on

cores_per_socket - the amount of cores_per_socket

instance_package_id - ID of the instance package

hot_add_cpu - true, if the CPU parameter can be changed without rebooting the VS, otherwise false

hot_add_memory - true, if the memory parameter can be changed without rebooting the VS, otherwise false

time_zone - the time zone set for the VS. This parameter is applicable only to Windows KVM and XEN virtual servers.

draas_mode - true if DRaaS is enabled for the Cloud; otherwise false

vapp_id - the ID of the vApp

vmware_tools - the ID of the VMware tools

vcenter_moref - the ID of the vCenter unit

template_version - version of the template

domain - specify the domain for this VS. The default value is localdomain. You can edit the default value for domain in /onapp/interface/config/on_app.yml. This parameter is not applicable for Windows virtual servers.

vcenter_reserved_memory - amount of RAM assigned to the vCenter VS

deleted_at - time when the VS was deleted

ip_addresses - an array of IP addresses with the following parameters:

- networking_ip_address - the address of the network
- id - the ID of the IP address
- address - IP address
- broadcast - broadcast address
- network_address - network address
- gateway - gateway address
- created_at type - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- updated_at type - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- user_id - the ID of the user this IP address is assigned to
- pxe type - true, if this address can be used for cloudbooting a compute resource
- hypervisor_id - the ID of a compute resource the IP address is associated with
- ip_range_id - the ID of the IP range from which the IP address should be assigned
- free - true if free, otherwise false
- netmask - netmask for the IP address

monthly_bandwidth_used - VS monthly bandwidth in KB

total_disk_size - the total disk size in GB of all disks assigned to the VS

support_incremental_backups - 1, if the VS supports incremental backups, and 0 if it does not.
OnApp Cloud 6.3 API Guide

CPU Priority - this is a new parameter reserved for further use; currently will have the same value as CPU Shares

Built from ISO - true if the VS is built from ISO; otherwise false

Built from OVA - true if the VS is built from ISO; otherwise false

Acceleration - true if the VS is built from ISO; otherwise false

Acceleration Status - the status of acceleration: active or inactive.

Hypervisor Type - the type of the compute resource the VS is built on (for example: Xen, KVM, VCloud, VMware)

virsh_console - set true to enable Virsh console for the VS, otherwise, false

Page History
v. 6.1 Edge 2
- added the virsh_console parameter

90.8 Add Instance Package VS

To add a preconfigured VS, use the following request:

POST /virtual_machines.xml
POST /virtual_machines.json

This section describes the API request that adds a VS using instance packages. For information on the API request that adds a VS and sets its resources, refer to Add VS.

XML Request Example

curl -1 -X POST -H 'Accept: application/xml' -H 'Content-Type: application/xml' -u user:userpass -d '<virtual_machine><template_id>8</template_id><label>zaza</label><hostname>zaza</hostname><initial_root_password>ehgebhewvtwh</initial_root_password><instance_package_id>3</instance_package_id><location_group_id>12</location_group_id><licensing_server_id>38</licensing_server_id><licensing_type>kms</licensing_type><licensing_key>keyexample</licensing_key><required_ip_address_assignment>1</required_ip_address_assignment><required_automatic_backup>0</required_automatic_backup><required_virtual_machine_build>1</required_virtual_machine_build><required_virtual_machine_startup>1</required_virtual_machine_startup><time_zone>Atlantic Time (Canada)</time_zone><enable_autoscale>0</enable_autoscale><custom_recipe_variables><custom_recipe_variable><name>varname</name><value>value</value><enabled>1</enabled><recipe_ids type='array'><recipe_id>11</recipe_id></recipe_ids></custom_recipe_variable></custom_recipe_variables></virtual_machine>' --url http://onapp.test/virtual_machines.xml

JSON Request Example

curl -1 -X POST -H 'Accept: application/json' -H 'Content-Type: application/json' -u user:userpass -d '{"virtual_machine": {"template_id": 8, "label": "zaza", "hostname": "zaza", "initial_root_password": "ehgebhewvtwh", "instance_package_id": 3, "location_group_id": 12, "licensing_server_id": 38, "licensing_type": "kms", "licensing_key": "keyexample", "required_ip_address_assignment": 1, "required_automatic_backup": 0, "required_virtual_machine_build": 1, "required_virtual_machine_startup": 1, "time_zone": "Atlantic Time (Canada)"}, "enable_autoscale": false, "custom_recipe_variables": {"varname": "value"}, "recipe_ids": ["11"]}' --url http://onapp.test/virtual_machines.json
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"virtual_machine": {"template_id": "8", "label": "zaza", "hostname": "zaza", "licensing_server_id": "38", "licensing_type": "kms", "licensing_key": "keyexample", "initial_root_password": "tyrshghj657th", "instance_package_id": "3", "location_group_id": "12", "required_ip_address_assignment": "1", "required_automatic_backup": "0", "required_virtual_machine_build": "1", "required_virtual_machine_startup": "1", "time_zone": "Atlantic Time (Canada)"}, "enable_autoscale": "0", "recipe_ids": ["11"], "custom_recipe_variables": {"custom_recipe_variable": {"name": "varname", "value": "var_value", "enabled": "1"}}}' --url http://onapp.test/virtual_machines.json

Where:

hostname* - set the host name for this VS
instance_package_id - ID of the instance package that will be used to build the VS
label* - user-friendly VS description
location_group_id - set the ID of the location group where the VS should be created
type_of_format - type of filesystem - ext4. For Linux templates, you can choose ext4 file system instead of the ext3 default one
required_automatic_backup - set 1 if you need automatic backups
required_virtual_machine_build * - set 1 to build VS automatically
required_virtual_machine_startup - set 1 to start up the VS automatically, otherwise set 0 (default state is "1")
time_zone - the time zone set for the VS. This parameter is applicable only to Windows virtual servers.

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows VS manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a VS after starting it if time synchronization is not completed for some reason.

required_ip_address_assignment* - set "1" if you want VS to be created with already assigned IP address, otherwise set "0"; IP address can be assigned after VS creation using this API request
admin_note - enter a brief comment for the VS. Optional parameter
note - a brief comment a user can add to a VS
template_id* - the ID of a template from which a VS should be built
licensing_server_id* - the ID of a template group where the KMS server details are indicated and to which the template belongs (either directly or through the child group). This parameter is for Windows virtual machines with KMS licensing type only
licensing_type* - the type of a license: mak, kms or user own license. This parameter is required for Windows virtual machines only
licensing_key* - the key of a license, required if you have selected own licensing type, and not required for MAK and KMS licensing types
initial_root_password - the root password for a VS. Optional, if none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _], and the following special characters: ~ ! @ $ * _ + \{ \} : ; ' , . ? / . You can use both lower- and uppercase letters.

The following characters are not allowed for Windows-based virtual servers:

- percent sign [%]
- double quotation marks ['"]
- brackets [<,>]
- vertical bar [|]
- caret ['^']
- ampersand [&]
- parentheses [(,)]

initial_root_password_encryption_key - specify the password encryption passphrase

recipe_ids - an array of recipe ID you want to run on the virtual server provisioning

custom_variables - an array of custom variables with the following details:

- enabled - true, if the variable is enabled, otherwise false
- id - variable ID
- name - variable name
- value - variable value script

90.9 Add VMware VS

Virtual servers running on VMware compute resources are managed exactly the same as common virtual servers. The only difference is the creation process.

Currently the use of IPv6 is not supported for VMware virtual servers.

To create a VMware virtual server, use the following request:

POST /virtual_machines.xml
POST /virtual_machines.json

XML Request Example
## JSON Request Example

```
curl -i -X POST -d '{"virtual_machine":{"template_id":"267","licensing_key":"","label":"zaza_ware_json","hostname":"zaza","hypervisor_group_id":"72","hypervisor_id":"29","initial_root_password":"qwaszx","initial_root_password_confirmation":"qwaszx","cpu_shares":"1","cpu_shares":1,"memory":"128","cpus":"1","primary_disk_size":"25","data_store_group_swap_id":"84","swap_disk_size":"0","customer_network_id":63,"required_automatic_backup":0,"required_virtual_machine_build":1,"required_virtual_machine_startup":1,"required_ip_address_assignment":1,"custom_recipe_variable":{"name":"varname","value":"var_value","enabled":1}}}' -u user:userpass http://onapp.test/virtual_machines.json -H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

- **memory** - amount of RAM assigned to the VS
- **cpus** - number of CPUs assigned to the VS
- **cpu_shares** - required parameter. For KVM compute resource the CPU priority value is always 100. For XEN, set a custom value. The default value for XEN is 1
- **hostname** - set the host name for this VS
- **label** - user-friendly VS description
- **primary_disk_size** - set the disk space for this VS
- **swap_disk_size** - set swap space. There is no swap disk for Windows-based VSs
- **type_of_format** - type of filesystem - ext4. For Linux templates, you can choose ext4 file system instead of the ext3 default one
- **data_store_group_primary_id** - set the ID of the data store zone to which this primary disk is allocated
- **data_store_group_swap_id** - set the ID of the data store zone to which this swap disk is allocated
- **primary_network_id** - the ID of the primary network. Optional parameter that can be used only if it is assigned to the network zone
primary_network_group_id - the ID of the primary network group. Optional parameter

required_automatic_backup - set 1 if you need automatic backups

rate_limit - set max port speed. Optional parameter: if none set, the system sets port speed to unlimited

required_virtual_machine_build* - set 1 to build VS automatically

required_virtual_machine_startup - set 1 to start up the VS automatically, otherwise set 0 (default state is "1")

required_ip_address_assignment* - set "1" if you want IP address to be assigned automatically after creation. Otherwise set "0"

admin_note - enter a brief comment for the VS. Optional parameter

note - a brief comment a user can add to a VS

template_id* - the ID of a template from which a VS should be built

licensing_server_id * - the ID of a template group where the KMS server details are indicated and to which the template belongs (either directly or through the child group). This parameter is for Windows virtual machines with KMS licensing type only

licensing_type * - the type of a license: mak, kms or user own license. This parameter is required for Windows virtual machines only

licensing_key * - the key of a license, required if you have selected own licensing type, and not required for MAK and KMS licensing types

hypervisor_group_id - the ID of the compute zone in which the VS will be created. Optional: if no compute zone is set, the VS will be built in any available compute zone

hypervisor_id - the ID of a compute resource where the VS will be built. Optional: if no compute resource ID is specified, the VS will be built on the compute resource with the least available RAM (but sufficient RAM for the VS)

initial_root_password - the root password for a VS. Optional, if none specified, the system will provide a random password. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _ ], and the following special characters: ~ ! @ # $ * _ + = \ { } [ ] : ; ' , . ? / . You can use both lower- and uppercase letters.

NOTE: It is not possible to set VS password when creating a Windows-based VMware virtual server without running a sysprep.

custom_variables - an array of custom variables with the following details:

- enabled - true, if the variable is enabled, otherwise false
- id - variable ID
- name - variable name
- value - variable value script

Page History

v.3.1

- added the following parameters:
  - custom_variables
  - enabled
  - id
90.10 View Encrypted VS Password

If the VS was created with password encryption enabled, you can use the following API call to view the password (the request returns the decrypted password).

To view the encrypted password, use the following request:

GET /virtual_machines/:id/with_decrypted_password.xml
GET /virtual_machines/:id/with_decrypted_password.json

XML Request Example:

curl -X GET -u user:userpass
http://onapp.test/virtual_machines/12/with_decrypted_password.xml?initial_root_password_encryption_key=encryptionkey

JSON Request Example:

curl -X GET -u user:userpass
http://onapp.test/virtual_machines/12/with_decrypted_password.json?initial_root_password_encryption_key=encryptionkey

Where:

id – the virtual server's ID

90.11 Build or Rebuild VS

Note that rebuild is not supported for OVA virtual servers.

To build or rebuild a VS, use the following request:

POST /virtual_machines/:virtual_machine_id/build.xml
POST /virtual_machines/:virtual_machine_id/build.json

XML Request Example

curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
'<virtual_machine><template_id>1</template_id><required_startup>1</required_startup></virtual_machine>' --url
http://onapp.test/virtual_machines/12/build.xml

JSON Request Example
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"virtual_machine":{"template_id":"1","required_startup":"1"}}' --url http://onapp.test/virtual_machines/12/build.json

Where:

template_id* - the ID of a template from which a VS should be built.

required_startup - set to 1 if you wish to start a VS after it is built. Otherwise set to 0.

Instead of virtual server ID (:virtual_machine_id) you may use virtual server identifier (:virtual_machine_identifier).

initial_root_password_encryption_key - specify the password encryption passphrase for Windows you should specify the licensing type:

licensing_type - the type of a license: mak, kms or user own license

licensing_key - the key of a license, required if you have selected OWN licensing type, and not required for MAK and KMS licensing types

licensing_server_id - the ID of a template group where the KMS server details are indicated and to which the template belongs (either directly or through the child group)

90.12 Edit VS

To edit a virtual server, use the following request:

PUT /virtual_machines/:id.xml
PUT /virtual_machines/:id.json

XML Request Example

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<virtual_machine><label>Test_API_Edit</label><memory>512</memory><min_iops>600</min_iops><cpu_shares>40</cpu_shares><cpus>4</cpus><allow_migration>1</allow_migration><allow_cold_resize>1</allow_cold_resize><time_zone>Atlantic Time (Canada)</time_zone><cpu_socket>2</cpu_socket><cpu_topol>1</cpu_topol></virtual_machine>' --url http://onapp.test/virtual_topology/l/cpusocket/2/tempoinfo/12.xml

JSON Request Example

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"virtual_machine":{"label":"Test_API_Edit","memory":"512","min_iops":"600","cpu_shares":"40","cpus":"4","allow_migration":"1", "allow_cold_resize": "1", "time_zone": "Atlantic Time (Canada)", "cpu_socket": "2", "cpu_topol": "1"}}' --url http://onapp.test/virtual_topology/l/cpusocket/2/tempoinfo/12.json

Where:
**label** - the VS name

**allow_migration** - set 1 to migrate a VS to a compute resource with sufficient resources if a compute resource has insufficient space to resize. Otherwise, set 0.

**allow_cold_resize** – set 1 to switch to cold resize when hot resize failed

**time_zone** - the time zone set for the VS. This parameter is applicable only to Windows virtual servers.

**cpu_sockets** - the number of CPU sockets

**cpu_topology** - set 1 to tie two or more vCPUs into a single socket. Otherwise, set 0.

If you want to enable topology, make sure you indicate the number of **cpu_sockets** and **cpus** for the topology to work properly.

After you edit the server's time zone, you need to stop and then start up the VS.

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows VS manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a VS after starting it if time synchronization is not completed for some reason.

**For virtual servers built by selecting resources manually:**

**memory** - the amount of RAM allocated to this VS in Mb

**min_iops** - the minimum number of IO operations per second

**cpus** - the number of CPUs of this VS

**cpu_shares** - CPU priority percentage

**cpu_units** - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket.

**For virtual servers built using instance packages:**

**instance_package_id** - ID of the new instance package

You can only choose from those instance packages that offer more disk size than the current instance package.

After you select a new instance package you can use the extra disk size to [create a new disk](index.html) for the VS or [make the existing VS disk larger](index.html).

If the VS is modified successfully, an HTTP 204 response is returned. If scheduling for changes fails, an HTTP 422 response is returned.
The `primary_disk_min_iops` and `swap_disk_min_iops` parameters are now redundant. Instead, use the `min_iops` parameter for Edit Disk request.

Page History

**v.6.0**
- added `min_iops` parameter

**v.5.8**
- added `cpu_sockets` and `cpu_topology` parameters

**v.4.1**
- added `instance_package_id` parameter

**v.4.0**
- removed `primary_disk_min_iops` and `swap_disk_min_iops` parameters

**v.3.3**
- added `cpu_units` parameter

### 90.13 Clone Virtual Server

You can [clone virtual servers](#) that run on Xen and KVM compute resources, including virtual servers built from OVA and ISO templates.

To clone a virtual server, run the following request:

```
POST /virtual_machines/:virtual_machine_id/clone.xml
```

**XML Request Example**

```
```

**JSON Request Example**

```
```
90.14 Change VS Owner

To reassign a VS to another user, use the following request:

POST /virtual_machines/:virtual_machine_id/change_owner.xml
POST /virtual_machines/:virtual_machine_id/change_owner.json

XML Request Example


JSON Request Example


Where:

user_id* – input ID of a new VS owner

custom_recipes_action - select one of the following options for virtual server's recipes:

- none - recipes owner will not be changed
- move - recipes owner will be changed
- copy - recipes will be copied to new virtual servers owner

backups_action - select one of the following options for virtual server's backups:

- none - backup owner will not be changed
- move - backup owner will be changed

- Instead of virtual server ID (:virtual_machine_id) you may use virtual server identifier (:virtual_machine_identifier).
- If VS can not be reassigned to another user, you will get an error message: "Errors: New owner has reached his backup creation limit or doesn't have enough disk space."

90.15 Reset VS Root Password

To reset the VS root password, use the following request:

POST /virtual_machines/:virtual_machine_id/reset_password.xml
POST /virtual_machines/:virtual_machine_id/reset_password.json
XML Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/reset_password.xml
-d
"<virtual_machine><initial_root_password>qwaszx321</initial_root_password>
<initial_root_password_encryption_key>property321</initial_root_password_encryption_key></virtual_machine>" -H 'Accept:application/xml' -H 'Content-type:application/xml'

JSON Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/reset_password.json
-d
'{"virtual_machine":{"initial_root_password":"qwaszx123",
"initial_root_password_encryption_key":"property"}}' -H 'Accept:application/json' -H 'Content-type:application/json'

Where:

virtual_machine_id* - id of the VS, for which you want to reset password.

initial_root_password - the new root password for a VS. It can consist of 6-32 characters, letters [A-Za-z], digits [0-9], dash [-] and lower dash [ _ ]. You can use both lowercase and uppercase letters.

The following characters are not allowed for Windows-based virtual servers:

- percent sign [%]
- double quotation marks ["]
- brackets [<,>]
- vertical bar [|]
- caret [^]
- ampersand [&]
- parentheses [()]

initial_root_password_encryption_key - specify the password encryption passphrase.

You can also reset a VS password using the OnApp 2.3.2 API request:

XML Request example

curl -i -X POST -H 'Accept: application/xml' -H 'Content-type:application/xml' -u user:userpass --url
http://onapp.test/virtual_machines/12/reset_password

JSON Request example

curl -i -X POST -H 'Accept: application/json' -H 'Content-type:application/json' -u user:userpass --url
http://onapp.test/virtual_machines/12/reset_password

Where:

virtual_machine_id* - id of the VS, for which you want to reset password.
90.16 Set SSH Keys

Note that the following API request will be deprecated in the future OnApp releases.

To assign SSH keys of all administrators and a VS owner to a virtual server, use the following request:

POST /virtual_machines/:virtual_machine_id/set_ssh_keys.xml
POST /virtual_machines/:virtual_machine_id/set_ssh_keys.json

XML Request Example

curl -X POST -u user:userpass

JSON Request Example

curl -X POST -u user:userpass

90.16.1 New API Request

To assign SSH keys of all administrators and a VS owner to a virtual server, use the following request:

PUT /virtual_machines/:virtual_machine_id/set_ssh_keys.xml
PUT /virtual_machines/:virtual_machine_id/set_ssh_keys.json

XML Request Example

curl -X PUT -u user:userpass

JSON Request Example

curl -X PUT -u user:userpass
90.17 CPU Quota

CPU quota is a percentage value limiting maximal VS CPU load. You can view and set the default value of CPU quota on the compute resource level and edit the custom value on the virtual server level.

90.17.1 Get CPU Quota for Virtual Server

This feature is available only for KVM compute resources.

To view CPU Quota for the virtual server, use the following request:
GET /virtual_machines/:virtual_machine_id/cpu_quota.xml
GET /virtual_machines/:virtual_machine_id/cpu_quota.json

XML Request Example


XML Output Example

```xml
<cpu_quota>
    <enabled type="boolean">false</enabled>
    <value type="integer">0</value>
</cpu_quota>
```

Where:

- **enabled** - true, if CPU Quota is enabled, otherwise, false
- **value** - the value of CPU Quota. The maximum value is 99%.

90.17.2 Edit CPU Quota for Virtual Server

This feature is available only for KVM compute resources.

To edit CPU Quota for the virtual server, use the following request:
PUT /virtual_machines/:virtual_machine_id/cpu_quota.xml
PUT /virtual_machines/:virtual_machine_id/cpu_quota.json
XML Request Example

<cpu_quota>
  <enabled>true</enabled>
  <value>2</value>
</cpu_quota>'

JSON Request Example

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' http://onapp.test/virtual_machines/12/cpu_quota.json -u admin:password -d '{"cpu_quota": {"enabled": true, "value": 19}}'

XML Output Example

```xml
<cpu_quota>
  <enabled type="boolean">true</enabled>
  <value type="integer">19</value>
</cpu_quota>
```

Where:

- **enabled** - true, if CPU Quota is enabled, otherwise, false
- **value** - set the value of CPU Quota. The maximum value is 99%.

### 90.18 Edit FQDN

To edit fully qualified domain name (FQDN), use the following request:

```
PATCH /virtual_machines/:virtual_machine_id/fqdn.xml
PATCH /virtual_machines/:virtual_machine_id/fqdn.json
```

XML Request Example

curl "http://onapp.test/virtual_machines/pwwdzgmqommwhe/fqdn.xml" -d "
<virtual-machine>
  <hostname>testhostname</hostname>
  <domain>testlocaldomain</domain>
</virtual-machine>"
-X PATCH \
-u user:pass \
-H "Accept: application/xml" \
-H "Content-Type: application/xml"

JSON Request Example
curl "http://onapp.test/virtual_machines/cqhfsoprsqnn/fqdn.json" -d 
{"virtual_machine":{"hostname":"testhostname","domain":"testlocaldomain"}} 
-X PATCH \ 
-u user:pass \ 
-H "Accept: application/json" \ 
-H "Content-Type: application/json"

Where:

hostname - the name of your host
domain - the name of your domain
force - true, if FQDN update is enabled in case the transaction fails with a running virtual server, otherwise, false
shutdown_type - select the shutdown type from the dropdown list (can be Gracefully shutdown or power off).
required_startup - set 1 to start up the virtual server automatically after the FQDN is updated. Otherwise, set 0

90.19 Migrate VS

To migrate a VS to another compute resource inside one compute resource zone, use the following request:

POST /virtual_machines/:virtual_machine_identifier/migration.json

JSON Request Example

curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d 
{"virtual_machine":{"destination":"1","cold_migrate_on_rollback":"1"}}' - 
-url http://onapp.test/virtual_machines/:virtual_machine_identifier/migration.j

Where:

destination* - the ID of a target compute resource where you migrate a VS
cold_migrate_on_rollback - set to 1 if you wish to switch to a cold migration if the hot migration fails, otherwise, set 0

Page History

v.5.5
- removed deprecated API requests:
  - POST /virtual_machines/:virtual_machine_id/migrate.xml
  - POST /virtual_machines/:virtual_machine_id/migrate.json

v.5.4
- added API request for OnApp 5.4 and up
90.20 Full Migrate VS

To migrate a VS with disks and/or NICs between compute resources with local storage or across compute zones, use the following request:

POST /virtual_machines/:virtual_machine_identifier/migration.json

**JSON Request Example**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/virtual_machines/:virtual_machine_identifier/migration.json -d '{"virtual_machine": {"migration_type": "hot_full", "cold_migrate_on_rollback": "0", "destination": {"hypervisor_group_id": "4", "hypervisor_id": "1", "disks_destinations": {"disk_id_N1": "data_store_id", "disk_id_N2": "data_store_id"}, "networks_destinations": {"801": {"network_id": 2, "ip_net_id": 1, "ip_range_id": 8, "ip_address": "10.30.0.0"}, "802": {"network_id": 108, "ip_net_id": 63, "ip_range_id": 49, "ip_address": "10.30.0.1"}}}}}'
```

Where:

- `migration_type` - set to `hot_full` if you want to run the hot migration. Set `full` if you want to run the cold migration.
- `cold_migrate_on_rollback` - set to 1 if you want to switch to the cold migration if the hot migration fails, otherwise, set 0. The parameter is not applicable if you already run the cold migration.
- `destination` - the array where you can pass an ID of destinations where you migrate a VS:
  - `hypervisor_group_id` - the ID of a target compute zone where you migrate a VS
  - `hypervisor_id` - the ID of a target compute resource where you migrate a VS
- `disks_destinations` - the array where you can pass the following parameters:
  - `disk_id` - the ID of the disk that you want to migrate
  - `data_store_id` - the ID of the target data store
- `networks_destinations` - the array where you can pass the following parameters:
  - `nic_id` - the ID of the NIC that you want to migrate
  - `network_id` - the ID of the destination network
  - `ip_net_id` - the ID of the destination IP Net
  - `ip_range_id` - the ID of the destination IP Range
  - `ip_address` - the IP address to assign to a virtual server after the migration

**Page History**

v.6.1 Edge 2

- added the following parameters:
  - `networks_destinations`
  - `nic_id`
  - `network_id`
  - `ip_net_id`
  - `ip_range_id`
### v.5.8
- Added the following parameters:
  - `migration_type`
  - `cold_migrate_on_rollback`
  - `data_store_id`
  - `virtual_machine_identifier`

### v.5.5
- Removed deprecated API requests:
  - POST `/virtual_machines/:virtual_machine_id/migrate.xml`
  - POST `/virtual_machines/:virtual_machine_id/migrate.json`

## 90.21 Migrate Multiple Virtual Servers

You can migrate multiple virtual servers at once from one compute resource to another compute resource of the same type (KVM to KVM or Xen to Xen). The mass migration is available within compute resources that belong to the same compute zone.

To migrate virtual servers, use the following request:

```plaintext
POST /hypervisors/:hypervisor_id/virtual_machines/migration.xml
POST /hypervisors/:hypervisor_id/virtual_machines/migration.json
```

### XML Request Example

```bash
```

### JSON Request Example

```bash
```

**Where:**

- `virtual_machines` - the array of virtual servers to migrate
- `virtual_machines_identifiers` - the array of virtual servers identifiers
- `virtual_machine_identifier` - the identifier of a virtual server
- `destination_hypervisor_id` - the ID of a destination compute resource
cold_migrate_on_rollback - set 1 if you want to apply cold migration in case of the hot migration failure, otherwise, set 0

If some of the selected virtual servers have disks that run as a local storage on this compute resource, these virtual servers could not be migrated. After the migration, these virtual servers remain on the previous compute resource, while other VSs are migrated to the destination compute resource.

### 90.22 Hot Migrate Disks

To hot migrate disks, use the following request:

**POST**
/virtual_machines/virtual_server_identifier/disks/:id/migration.xml

**POST**
/virtual_machines/virtual_server_identifier/disks/:id/migration.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**Where:**

- **type** - the type of migration
- **data_store_id** - the ID of the data store
- **virtual_machine_id** - the ID of the virtual server
- **disk_id** - the ID of the disk

### 90.23 Migrate VS from Xen to KVM

To migrate a Linux-based VS from Xen to KVM compute resource, use the following request:

**POST** /virtual_machines/:virtual_machine_id/migration.xml

**POST** /virtual_machines/:virtual_machine_id/migration.json
### XML Request Example

```
```

### JSON Request Example

```
```

Where:

- **migration_type** - set to `xen_to_kvm` if you want to run the migration from Xen to KVM.
- **backup_before_migration** - set to 1 if you want to schedule a backup before the migration. Otherwise, set to 0 (default state is "1")
- **delete_old_backups** - set to 1 if you want to remove old backups incompatible with KVM after the migration is completed. Otherwise, set to 0 (default state is "1")

If enabled, this option deletes all the old backups besides the one that is taken right before the migration provided that you also enabled scheduling a backup before the migration.

- **backup_after_migration** - set to 1 if you want to schedule backups in case of successful migration. Otherwise, set to 0 (default state is "1")
- **destination** - hash of destination parameters:
  - **hypervisor_group_id** - the ID of a target compute zone where you migrate a VS
  - **hypervisor_id** - the ID of a target compute resource where you migrate a VS
  - **disks_destinations** - the array where you can pass the following parameters:
    - **disk_id** - the ID of the disk that you want to migrate
    - **data_store_id** - the ID of the target data store

### 90.24 Set VIP Status for VS

To set/remove VIP status for a VS, use the following request:

```bash
POST /virtual_machines/:id/set_vip.xml
POST /virtual_machines/:id/set_vip.json
```
### XML Request Example

```
```

### JSON Request Example

```
```

Where:

`vip` - whether VIP status is enabled for the server or not. Set this parameter to 'true' to enable and to 'false' to disable the VIP status.

---

### 90.25 Delete VS

To delete a VS, use the following request:

```bash
DELETE /virtual_machines/:id.xml
DELETE /virtual_machines/:id.json
```

To delete a virtual server together with its backups, the user needs to have the *Destroy any backup* or *Destroy own backup* permission enabled. Otherwise, the backups of the VS deleted by the user will remain in the system.

#### XML Request Example

```
curl -i -X DELETE -u user:userpass http://onapp.test/virtual_machines/12.xml?convert_last_backup=1&destroy_all_backups=1
```

#### JSON Request Example

```
curl -i -X DELETE -u user:userpass http://onapp.test/virtual_machines/12.json?convert_last_backup=1&destroy_all_backups=1
```

Where:

`id` – the ID of a VS you want to delete

`convert_last_backup` – set 1 to convert the last VS’s backup to template, otherwise set 0

`destroy_all_backups` – set 1 to destroy all existing backups of this VS, otherwise set 0
90.26 Start up VS

When you start up a VS, it might be implicitly cold migrated if the current compute resource does not have sufficient resources. For more information, refer to Virtual Server Provisioning.

To start up a VS, use the following request:

POST /virtual_machines/:virtual_machine_id/startup.xml
POST /virtual_machines/:virtual_machine_id/startup.json

XML Request Example

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/startup.xml
```

JSON Request Example

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/startup.json
```

You can also start up a VS in recovery mode. For this, run the following request:

XML Request Example

```
curl -i -X POST -u user:userpass -d '<mode>recovery</mode>' --url
http://onapp.test/virtual_machines/12/startup.xml
```

JSON Request example

```
curl -i -X POST -u user:userpass -d '{"mode":"recovery"}' --url
http://onapp.test/virtual_machines/12/startup.json
```

90.27 Segregate VS

To segregate a VS (that is, instruct it never to reside on the same compute resource as another VS), use the following request:

PUT /virtual_machines/:virtual_machine_id/segregation.xml
PUT /virtual_machines/:virtual_machine_id/segregation.json

XML Request Example

```
curl -i -X POST -u user:userpass -d '<mode>recovery</mode>' --url
http://onapp.test/virtual_machines/12/startup.xml
```

JSON Request example

```
curl -i -X POST -u user:userpass -d '{"mode":"recovery"}' --url
http://onapp.test/virtual_machines/12/startup.json
```

To segregate a vs (that is, cancel the instruction for it to never reside on the same compute resource as another vs), use the following request:

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<virtual_machine><strict_virtual_machine_id>123</strict_virtual_machine_id></virtual_machine>' --url
  http://onapp.test/virtual_machines/12/segregation.xml
```

**JSON Request Example**

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"virtual_machine":{"strict_virtual_machine_id":"123"}}' --url
  http://onapp.test/virtual_machines/12/segregation.json
```

**Where:**

`strict_virtual_machine_id` * - the ID of virtual server you wish to segregate from the given VS

---

**90.28 Desegregate VS**

To desegregate a VS (that is, cancel the instruction for it to never reside on the same compute resource as another VS), use the following request:

```
curl -i -X DELETE -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<virtual_machine><strict_virtual_machine_id>123</strict_virtual_machine_id></virtual_machine>' --url
  http://onapp.test/virtual_machines/12/segregation.xml
```

**XML Request Example**

```
curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"virtual_machine":{"strict_virtual_machine_id":"123"}}' --url
  http://onapp.test/virtual_machines/12/segregation.json
```

**JSON Request Example**

```
curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"virtual_machine":{"strict_virtual_machine_id":"123"}}' --url
  http://onapp.test/virtual_machines/12/segregation.json
```

**Where:**

`strict_virtual_machine_id` * - the ID of virtual server you wish to desegregate from the given VS
90.29 Reboot VS

To reboot a VS, use the following request:

POST /virtual_machines/:virtual_machine_id/reboot.xml
POST /virtual_machines/:virtual_machine_id/reboot.json

XML Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/reboot.xml

JSON Request Example

curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/reboot.json

An HTTP 201 response is returned on a successful reboot. Unsuccessful reboot responses include HTTP 404 (resource not found – e.g. if the VS isn’t online) and HTTP 422 (request cannot be processed – for example, if parameters were incorrect).

90.30 Get List of Blacklisted Domains

To view the list of blacklisted domains, use the following request:

GET /virtual_machines/:virtual_machine_id/blacklist_domains.xml
GET /virtual_machines/:virtual_machine_id/blacklist_domains.json

XML Request Example

curl -i -X GET

JSON Request Example

curl -i -X GET
http://onapp.test/virtual_machines/xungcyuakcyyeb/blacklist_domains.json
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
XML Output Example

```xml
<hash>
  <blacklist_domains type="array">
    <blacklist_domain>site4.com</blacklist_domain>
    <blacklist_domain>site5.com</blacklist_domain>
  </blacklist_domains>
</hash>
```

**Where:**

blacklist_domain - the label of the domain to be blacklisted from being accelerated

### 90.31 Edit Blacklisted Domains

To edit blacklisted domains, use the following request:

**PUT /virtual_machines/:virtual_machine_id/blacklist_domains.xml**

**PUT /virtual_machines/:virtual_machine_id/blacklist_domains.json**

**XML Request Example**

```
```

**JSON Request Example**

```
```

**Where:**

hostname_blacklist - the label of the domain to be blacklisted from being accelerated

### 90.32 Remove All Domains from Blacklist.

To remove all IP addresses from a blacklist, use the following request:

**PUT /virtual_machines/:virtual_machine_id/blacklist_domains.xml**

**PUT /virtual_machines/:virtual_machine_id/blacklist_domains.json**

**XML Request Example**

```
```
curl -i -X PUT
http://onapp.test/virtual_machines/xungcyuakcyeb/blacklist_domains.xml -d
'<?xml version="1.0"?><blacklist_domains><hostname_blacklists type="string"></hostname_blacklists></blacklist_domains>' -u user:userpass
-H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X PUT
http://onapp.test/virtual_machines/xungcyuakcyeb/blacklist_domains -d
'{"blacklist_domains":{"hostname_blacklists":""}}' -u user:userpass
-H 'Accept: application/json' -H 'Content-type: application/json'

90.33 Purge File(s)

This action is available only for accelerated virtual servers.

To purge one or several files, the system will compare the checksum of the cached file and the
new one. The cached file will only be purged if the checksums vary, that is, the files are
different. If the checksum of the two files are the same, the cached file will not be purged.

To purge one or several cached files, use the following request:

POST /virtual_machines/:virtual_machine_id/purge.xml
POST /virtual_machines/:virtual_machine_id/purge.json

XML Request Example

curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url

JSON Request Example

curl -i -X POST -u user:userpass -H 'Accept: application/json' -H
'Content-type: application/json'--url
'http://onapp.test/virtual_machines/12/purge.json' -d

Where:

purge_path - the path to the file you want to purge

If you need to purge all content, refer to Purge All Content.
90.34 Purge All Content

This action is available only for accelerated virtual servers.

To purge all cached content, use the following request:

POST /virtual_machines/:virtual_machine_id/purge_all.xml
POST /virtual_machines/:virtual_machine_id/purge_all.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

If you need to purge only certain files, refer to Purge File(s).

90.35 Reboot VS in Recovery

To reboot a VS in recovery mode with a temporary login ("root") and password ("recovery"), use the following request:

POST /virtual_machines/:virtual_machine_id/reboot.xml
POST /virtual_machines/:virtual_machine_id/reboot.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

90.36 Reboot VS from ISO

To boot a virtual server that is powered on from an ISO, use the following request:

POST /virtual_machines/:virtual_machine_id/reboot.xml
POST /virtual_machines/:virtual_machine_id/reboot.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/reboot.xml -d '<iso_id>11</iso_id>'
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/reboot.json -d '{"iso_id": "11"}'
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

*virtual_machine_id* - the ID of the VS you want to reboot

*iso_id* - ID of the ISO you want to use

### 90.37 Boot VS from ISO

To boot virtual servers that are powered off from an ISO, use the following request:

POST /virtual_machines/:virtual_machine_id/startup.xml

POST /virtual_machines/:virtual_machine_id/startup.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/startup.xml -d '<iso_id>11</iso_id>'
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/startup.json -d '{"iso_id": "11"}'
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

*virtual_machine_id* - the ID of the VS you want to boot

*iso_id* - the ID of the ISO you want to boot from

### 90.38 Suspend VS

To suspend a VS, use the following request:

POST /virtual_machines/:id/suspend.xml

POST /virtual_machines/:id/suspend.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/suspend.xml
```

**JSON Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/suspend.json
```

Where:

*virtual_machine_id* - the ID of the VS you want to suspend

*iso_id* - the ID of the ISO you want to suspend from
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/suspend.xml

**JSON Request Example**

curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/suspend.json

Where:

`virtual_machine_id*` - ID of a VS you want to suspend

### 90.39 Unlock VS

To unlock a VS, use the following request:

POST /virtual_machines/:virtual_machine_id/unlock.xml
POST /virtual_machines/:virtual_machine_id/unlock.json

**XML Request Example**

```bash
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/unlock.xml
```

**JSON Request Example**

```bash
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/unlock.json
```

### 90.40 Unsuspend VS

To activate a VS again, use the same request as to suspend it:

POST /virtual_machines/:id/suspend.xml
POST /virtual_machines/:id/suspend.json

For details, refer to the [Suspend a VS] section.

### 90.41 Shut down VS

To terminate the VS gracefully, use the following request:

POST /virtual_machines/:virtual_machine_id/shutdown.xml
POST /virtual_machines/:virtual_machine_id/shutdown.json

**XML Request Example**

```bash
curl -i -X POST -u user:userpass --url
http://onapp.test/virtual_machines/12/shutdown.xml
```
curl -i -X POST -u user:userpass --url 
http://onapp.test/virtual_machines/12/shutdown.xml

**JSON Request Example**

curl -i -X POST -u user:userpass --url 
http://onapp.test/virtual_machines/12/shutdown.json

90.42 Stop VS

To terminate the VS forcefully, use the following request:

POST /virtual_machines/:virtual_machine_id/stop.xml
POST /virtual_machines/:virtual_machine_id/stop.json

**XML Request Example**

curl -i -X POST -u user:userpass --url 
http://onapp.test/virtual_machines/12/stop.xml

**JSON Request Example**

curl -i -X POST -u user:userpass --url 
http://onapp.test/virtual_machines/12/stop.json

90.43 Open VS Console

To open a VS console:

1. Run the following request:
   
   GET /virtual_machines/:virtual_machine_id/console.xml
   GET /virtual_machines/:virtual_machine_id/console.json

2. Find and copy the value for the `remote_key` parameter in the response output.

3. Open the following URL in the browser:
   
   http://onapp.test/console_remote/[remote_key_parameter_value]

90.44 VS Autoscaling

VS autoscaling allows you to automatically increase the RAM, CPU and disk size of a virtual server. Disk usage autoscaling is applicable for VS primary disk only. VS resources autoscaling is based on the rules you specify. For example, you can set up a rule that will add 1000 MB of memory to a VS if RAM has been above 90% for the last 10 minutes - but add no more than 5000 MB in total in 24 hours.
90.44.1 Enable Autoscaling for VS

To enable autoscaling for a virtual server, use the following request:

POST /virtual_machines/:virtual_machine_id/autoscale_enable.xml
POST /virtual_machines/:virtual_machine_id/autoscale_enable.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

90.44.2 Get the List of Autoscaling Rules for VS

To get the list of autoscaling rules for a particular VS, use the following request:

GET /virtual_machines/:virtual_machine_id/auto_scaling.xml
GET /virtual_machines/:virtual_machine_id/auto_scaling.json

**XML Request Example**

```
curl -i -X GET -u user:userpass --url http://onapp.test/virtual_machines/3823/auto_scaling.xml
```

**JSON Request Example**

```
curl -i -X GET -u user:userpass --url http://onapp.test/virtual_machines/3823/auto_scaling.json
```

**XML Output Example**
<auto_scaling_configurations type="array">
  <auto_scaling_configuration>
    <adjust_units type="integer">10</adjust_units>
    <created_at type="datetime">2015-02-27T16:11:12+02:00</created_at>
    <for_minutes type="integer">5</for_minutes>
    <id type="integer">6</id>
    <limit_trigger type="integer">10</limit_trigger>
    <resource>cpu</resource>
    <scale_type>up</scale_type>
    <up_to type="integer">50</up_to>
    <updated_at type="datetime">2015-02-27T16:11:12+02:00</updated_at>
    <virtual_machine_id type="integer">3823</virtual_machine_id>
    <above type="integer">10</above>
    <add_units type="integer">10</add_units>
  </auto_scaling_configuration>
  ...
</auto_scaling_configuration>

Where:

- **adjust_units** - the amount of resource units which the system should add/remove if the rule is met
- **created_at** - the date when the record in DB was created
- **for_minutes** - the time threshold before scaling will be triggered
- **id** - the ID of the rule
- **limit_trigger** - the amount of the resource usage (%). If this value is reached by the VS for the period specified by the for_minutes parameter, the system will add/remove the amount of units set by the adjust_units parameters
- **resource** - the resource for which the rule is created (memory/cpu/disk)
- **scale_type** - the autoscale option for this rule: up or down
- **up_to** - the amount of resource which cannot be exceeded within 24 hours period
- **updated_at** - the date when the record in DB was updated
- **virtual_machine_id** - the ID of the VS to which this rule applies
- **above** - the amount of the resource usage (%). If this value is reached by the VS during the period specified by the for_minutes parameter, the system will add the amount of units set by the add_units parameters
- **add_units** - the amount of resource units which the system should add if the rule is met

### 90.44.3 Create Autoscaling Rule for VS

To create autoscaling rule for a virtual server, use the following request:

```
POST /virtual_machines/:virtual_machine_id/auto_scaling.xml
POST /virtual_machines/:virtual_machine_id/auto_scaling.json
```

**XML Request Example**
OnApp Cloud 6.3 API Guide

```
curl -i -X POST -u user:password
http://onapp.test/virtual_machines/12/auto_scaling.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<auto_scaling_configurations><up><cpu><enabled>1</enabled><for_minutes>5</for_minutes><limit_trigger>40</limit_trigger><adjust_units>20</adjust_units><up_to>100</up_to><memory><enabled>1</enabled><for_minutes>5</for_minutes><limit_trigger>128</limit_trigger><adjust_units>128</adjust_units><up_to>1024</up_to><disk><enabled>1</enabled><for_minutes>5</for_minutes><limit_trigger>80</limit_trigger><adjust_units>10</adjust_units><up_to>20</up_to></disk></memory></cpu></up><down><cpu><enabled>1</enabled><for_minutes>5<for_minutes><limit_trigger>100</limit_trigger><adjust_units>10</adjust_units></cpu><memory><enabled>1</enabled><for_minutes>5<for_minutes><limit_trigger>100</limit_trigger><adjust_units>128</adjust_units></memory><disk><enabled>1</enabled><for_minutes>5<for_minutes><limit_trigger>80</limit_trigger><adjust_units>10</adjust_units></disk></down></auto_scaling_configurations>'
```

**JSON Request Example**

```
curl -i -X POST -u user:password
http://onapp.test/virtual_machines/12/auto_scaling.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"auto_scaling_configurations":{"up":{"cpu":{"enabled":"1", "for_minutes":"5", "limit_trigger":"40", "adjust_units":"20", "up_to":"100"},"memory":{"enabled":"1", "for_minutes":"5", "limit_trigger":"90", "adjust_units":"128", "up_to":"1024"},"disk":{"enabled":"1", "for_minutes":"5", "limit_trigger":"80", "adjust_units":"10", "up_to":"20"}},"down":{"cpu":{"enabled":"1", "for_minutes":"5", "limit_trigger":"40", "adjust_units":"20"},"memory":{"enabled":"1", "for_minutes":"5", "limit_trigger":"90", "adjust_units":"128", "up_to":"1024"},"disk":{"enabled":"1", "for_minutes":"5", "limit_trigger":"80", "adjust_units":"10"}}}
```

Where:

- `up_to`* - the amount of resource which cannot be exceeded within 24 hours period; only for autoscale up rules
- `for_minutes`* - the time threshold before scaling will be triggered
- `limit_trigger`* - the amount of the resource usage (%). If this value is reached by the VS for the period specified by the `for_minutes` parameter, the system will add/remove the amount of units set by the `adjust_units` parameters.
- `adjust_units`* - the amount of resource units which the system should add/remove if the rule is met
- `enabled`* - set 1 to enable, or 0 to disable
- `cpu/memory/disk` - indicate type of resource for which the autoscaling rule is set (Disk usage autoscaling is applicable for VS primary disk only)

### 90.44.4 Edit Autoscaling Rule for VS

At present you cannot edit separate elements of autoscaling rule. To change a rule for a VS you have to create a new rule, using the same request as in Create Autoscaling Rule section.

### 90.44.5 Delete Autoscaling Rule

To delete autoscaling rules, use the following request:
DELETE /virtual_machines/:virtual_machine_id/auto_scaling.xml
DELETE /virtual_machines/:virtual_machine_id/auto_scaling.json

XML Request Example

```sh
curl -i -X DELETE -u user:userpass --url http://onapp.test/virtual_machines/12/auto_scaling.xml
```

JSON Request Example

```sh
curl -i -X DELETE -u user:userpass --url http://onapp.test/virtual_machines/12/auto_scaling.json
```

This will delete all autoscaling rules set for this VS.

90.44.6 Disable Autoscaling for VS

To disable autoscaling for a virtual server, use the following request:

POST /virtual_machines/:virtual_machine_id/autoscale_disable.xml
POST /virtual_machines/:virtual_machine_id/autoscale_disable.json

XML Request Example

```sh
```

JSON Request Example

```sh
```

90.45 Get VS Billing Statistics

You can view the billing statistics for a particular VM using the following request:

GET /virtual_machines/:virtual_machine_id/vm_stats.xml
GET /virtual_machines/:virtual_machine_id/vm_stats.json

To get a shorter statistics output, add an id parameter in the URL:

GET /virtual_machines/:virtual_machine_id/vm_stats/:vm_stats_id.xml
GET /virtual_machines/:virtual_machine_id/vm_stats/:vm_stats_id.json

Define a shorter period by setting Start and End time in the API call:

GET /virtual_machines/12/vm_stats.xml?period[startdate]=YYYY-MM-DD+hh:mm:ss&period[enddate]=YYYY-MM-
DD+hh:mm:ss&period[use_local_time]=1
GET /virtual_machines/12/vm_stats.json?period[startdate]=YYYY-MM-
DD+hh:mm:ss&period[enddate]=YYYY-MM-
DD+hh:mm:ss&period[use_local_time]=1

XML Output Example
<vm_hourly_stats type="array">
<vm_hourly_stat>
<created_at type="datetime">2011-08-09T12:00:10Z</created_at>
<currency_code>USD</currency_code>
<id type="integer">8248</id>
<stat_time type="datetime">2011-08-09T12:00:00Z</stat_time>
<updated_at type="datetime">2011-08-09T12:00:10Z</updated_at>
<user_id type="integer">1</user_id>
<virtual_machine_id type="integer">44</virtual_machine_id>
<vm_billing_stat_id type="integer">100175</vm_billing_stat_id>
<billing_state>...
<disks type="array">
<disk>
<id type="integer">2933</id>
<costs type="array">
<cost>
<value type="integer">5</value>
<cost type="float">3.0</cost>
<resource_name>disk_size</resource_name>
</cost>
</costs>
<label>Disk #2933</label>
</disk>
</disks>

<network_interfaces type="array">
<network_interface>
<id type="integer">2688</id>
<costs type="array">
<cost>
<value type="integer">1</value>
<cost type="float">0.0</cost>
<resource_name>ip_addresses</resource_name>
</cost>
<cost>
<value type="integer">1.0</value>
<cost type="float">100.0</cost>
<resource_name>instance_package</resource_name>
<resource_id>90</resource_id>
</cost>
<cost>
<value type="integer">1.0</value>
<cost type="float">0.0</cost>
<resource_name>template_usage</resource_name>
<resource_id>5</resource_id>
</cost>
</costs>
<label>eth0</label>
</network_interface>
</network_interfaces>
</service_addons type="array">
</service_addon>
</service_addons>
</virtual_machines type="array">
```xml
<virtual_machine>
  <id type="integer">1701</id>
  <costs type="array">
    <cost>
      <value type="integer">1</value>
      <cost type="float">0.0</cost>
      <resource_name>cpus</resource_name>
    </cost>
    <!-- costs for other resources go here -->
  </costs>
  <label>zaza_CP_3.2 (do not remove)</label>
</virtual_machine>
</virtual_machines>
</billing_stats>
</vm_hourly_stat>
</vm_hourly_stats>
```

Where:
- **created_at** – the timestamp in DB when this record was created
- **currency_code** - currency in which this virtual machine is charged within the bucket
- **id** – the ID of the server hourly statistics. You can add this parameter to the request URL to get a shorter statistics output.
- **stat_time** – the particular hour for which these statistics were generated
- **updated_at** – the date when these statistics were updated
- **user_id** - the ID of VS owner
- **virtual_machine_id** - the ID of the VS
- **vm_billing_stat_id** - billing statistics ID
- **billing_stats** - an array of billing details for the resources used by this VM

When generating billing statistics, OnApp takes the last state of the VS during the hour. For example, if a VS was turned on at 6.15 and turned off at 6.59 it will be considered as being off for the whole hour and its resources will be billed according to the OFF prices set in the bucket. However, the VS’s disk and network interface usage can still be billed in case the VS was on during that hour.

**disks** - an array of disks used by this VM with their billing details:
- **id** - disk ID used in database
- **costs** - an array of disk related resources with their total prices for the period specified in the stat-time parameter, where:
  - **value** - the amount of resources used (GBs of disk size, Kbs of data read/written, the number of reads/writes)
  - **cost** - the total due for the resource
  - **resource_name** - the resource in question. This can be disk_size, data_read, data_written, reads_completed and writes_completed
label - disk name used in UI

network_interfaces - an array of network interfaces used by this VM with their billing statistics:
  id - network interface ID
  costs - an array of network interface related resources with their total prices for the period specified in the stat-time parameter, where:
    value - the amount of resources used by this network interface (the number of IPs, the port speed in Mb per second, the data sent and received in KBs)
    cost - the total due for the resource
  resource_name - the resource in question. This can be ip_addresses, rate, data_received and data_sent
  label - network interface name used in OnApp

service_addons - an array of service add-ons assigned to this VS with their billing details:
  id - service add-on ID
  costs - an array of service add-on related resources with their total prices for the period specified in the stat-time parameter, where:
    value - the amount of resources used by this service add-on (at the moment value will always be "1")
    cost - the total due for the resource
  resource_name - the resource in question
  label - service add-on name used in OnApp

virtual_machine - an array of virtual machine billing details:
  id - virtual server ID
  costs - an array of VS resources with their total prices for the period specified in the stat-time parameter, where:
    value - the amount of resources allocated to this VM. For the templates resource, this parameter means a template ID in database.
    cost - the total due for this resource
  resource_name - the resource in question. This can be cpu_shares, cpus, cpu_usage, cpu_time, memory, instance_package, and template_usage
  resource_id - the ID for the instance_package and template_usage resources
  label - the VS name

total_cost – the total amount of money owed for the VM specified by id parameter for a particular hour specified by stat_time parameter (total_cost = vm_resources_cost + usage_cost)

vm_resources_cost – the amount of money due for the VM resources for the particular hour specified by stat_time parameter (memory, disks, templates)

usage_cost – the total due for VM usage for this particular hour specified by stat_time parameter (data sent/received, bandwidth, CPU usage)

Page History
v. 6.0
  • added the resource_id parameter for the instance_package and template_usage resources
v. 5.3
  • added the service_addons array of parameters
90.46 Search VS by Label

To search virtual servers by label, use the following request:

GET /virtual_machines.xml?q=label
GET /virtual_machines.json?q=label

XML Request Example

```
```

JSON Request Example

```
```

Where you have to specify the label of a virtual server you are searching for.

90.47 Get VS CPU Usage Statistics

To view CPU usage statistics of a virtual server, use the following request:

GET /virtual_machines/:virtual_machine_id/cpu_usage.xml
GET /virtual_machines/:virtual_machine_id/cpu_usage.json

Define a shorter period by setting Start and End time in the API call:


XML Request Example

```
```

JSON Request Example

```
```

Where you have to specify the virtual server ID.

**XML Output Example**

```xml
<cpu_hourly_stats type="array">
  <cpu_hourly_stat>
    <cpu_time type="integer">18</cpu_time>
    <created_at type="datetime">2015-01-06T10:00:18Z</created_at>
    <stat_time type="datetime">2015-01-06T10:00:00Z</stat_time>
    <updated_at type="datetime">2015-01-06T10:00:18Z</updated_at>
    <user_id type="integer">1</user_id>
    <virtual_machine_id type="integer">1701</virtual_machine_id>
  </cpu_hourly_stat>
  ...<cpu_hourly_stat>
</cpu_hourly_stats>
```

Where:
- **cpu_time** - use the following formula to convert CPU data received in the API output:
  
  \[ \text{CPU} = \frac{\text{cpu_time}}{10} \times \frac{1}{3600} \]
  
  Where cpu_time is data from API output.
  
  For example: \( \text{cpu_time} = 2330 \), then: \( \frac{2330}{10 \times 3600} = 0.06 \) (6%).
  
  We use “cpu_time” * 10 to correct store fractional values.
- **created_at** - the timestamp in DB when this record was created
- **stat_time** - the particular hour for which these statistics were generated
- **updated_at** - the time stamp in DB when this record was updated
- **user_id** - the ID of the VS owner
- **virtual_machine_id** - ID of the VS

### 90.48 Add/Edit Admin/User Note for Virtual Server

To edit/make an admin note, use the following request:

PUT /virtual_machines/:virtual_machine_id.xml
PUT /virtual_machines/:virtual_machine_id.json

or

PUT /virtual_machines/:virtual_machine_id/admin_note.xml
PUT /virtual_machines/:virtual_machine_id/admin_note.json

**XML Request Example**

```bash
curl -i -X PUT -u user:userpass http://onapp.test/virtual_machines/12.xml
-d '<virtual_machine><admin_note>agfagwe tiuuytjgh
yuytu</admin_note></virtual_machine>' -H 'Accept:application/xml' -H 'Content-type:application/xml'
```
or

curl -i -X PUT -u user:userpass
http://onapp.test/virtual_machines/12/admin_note.xml -d
'"<virtual_machine><admin_note>agfagwe tiuuytjgh
yuytu</admin_note></virtual_machine>"' -H 'Accept:application/xml' -H
'Content-type:application/xml'

or

curl -i -X PUT -u user:userpass http://onapp.test/virtual_machines/12/json
-d '{"virtual_machine":{"admin_note":"kjfjhjtrtjt"}}' -H
'Accept:application/json' -H 'Content-type:application/json'

or

curl -i -X PUT -u user:userpass
http://onapp.test/virtual_machines/12/admin_note.json -d
'"{"virtual_machine":{"admin_note":"kjfjhjtrtjt"}}"' -H
'Accept:application/json' -H 'Content-type:application/json'

Where:

admin_note – enter the text of your note.

90.48.1 Add/Edit User Note

To edit/make a user note, use the following request:

PUT /virtual_machines/:virtual_machine_id.xml

or

PUT /virtual_machines/:virtual_machine_id.json

or

PUT /virtual_machines/:virtual_machine_id/note.xml

PUT /virtual_machines/:virtual_machine_id/note.json

XML Request Example

curl -i -X PUT -u user:userpass http://onapp.test/virtual_machines/12.xml
-d '<virtual_machine><note>agfagwe tiuuytjgh
yuytu</note></virtual_machine>' -H 'Accept:application/xml' -H 'Content-type:application/xml'

or

curl -i -X PUT -u user:userpass
http://onapp.test/virtual_machines/12/note.xml -d
'"<virtual_machine><note>agfagwe tiuuytjgh_yuytu</note></virtual_machine>"'
-H 'Accept:application/xml' -H 'Content-type:application/xml'
JSON Request Example

```
curl -i -X PUT -u user:userpass http://onapp.test/virtual_machines/12.json
   -d '{"virtual_machine":{"note":"kjfjhjtrjtj"}}' -H 'Accept:application/json' -H 'Content-type:application/json'
```

or

```
curl -i -X PUT -u user:userpass
   http://onapp.test/virtual_machines/12/note.json
   -d '{"virtual_machine":{"note":"kjfjhjtrjtj"}}' -H 'Accept:application/json'
   -H 'Content-type:application/json'
```

Where:

- **note** – enter the text of your note.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no virtual server with a requested ID, or URL is incorrect.

90.49 Enable Booting from CD for ISO Virtual Server

You can configure whether the VS built from ISO should be booted from the ISO template location (CD emulation) or from the disk where the VS is provisioned. To enable booting from CD for ISO VS, use the following request:

- POST /virtual_machines/:id/cd_boot/enable.xml
- POST /virtual_machines/:id/cd_boot/enable.json

If disabled, the VS will be booted from the disk.

**XML Request Example**

```
curl -i -X POST http://onapp.test/virtual_machines/12/cd_boot/enable.xml
   -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request Example**

```
curl -i -X POST http://onapp.test/virtual_machines/12/cd_boot/enable.json
   -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

90.50 Disable Booting from CD for ISO Virtual Server

You can configure whether the VS built from ISO should be booted from the ISO template location (CD emulation) or from the disk where the VS is provisioned. To disable booting from CD for ISO VS, use the following request:

- POST /virtual_machines/:id/cd_boot/disable.xml
- POST /virtual_machines/:id/cd_boot/disable.json
In this case, the VS will be booted from the disk where the VS is provisioned.

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

---

90.51 Get List of Service Add-ons Assigned to VS

To get the list of service add-ons assigned to the VS, use the following request:

GET /virtual_machines/:virtual_machine_id/service_addons.xml
GET /virtual_machines/:virtual_machine_id/service_addons.json

**XML Request Example**

```bash
curl "http://onapp.test/virtual_machines/12/service_addons.xml" -X GET -u user:password
```

**JSON Request Example**

```bash
curl "http://onapp.test/virtual_machines/12/service_addons.json" -X GET -u user:password
```
<service_addons type="array">
  <service_addon>
    <id type="integer">2</id>
    <label>service_addon2</label>
    <description nil="true"/>
    <compatible_with type="array">
      <compatible_with>unix</compatible_with>
    </compatible_with>
    <user_id type="integer">2</user_id>
    <icon>
      <url nil="true"/>
    </icon>
    <created_at type="dateTime">2016-12-20T16:19:39+00:00</created_at>
    <updated_at type="dateTime">2016-12-20T16:19:39+00:00</updated_at>
  </service_addon>
</service_addons>

Where:

id – ID of the service add-on

label – the service add-on title

description – description text added to the service add-on

compatible_with – the OS type, with which this service add-on is compatible

user_id – ID of the user, who created the service add-on

icon – URL with the service add-on icon

created_at – the date when the service add-on was created in the [YYYY][MM][DD][hh][mm][ss]Z format

updated_at — the date when the service add-on was updated in the [YYYY][MM][DD][hh][mm][ss]Z format

90.52 Assign Service Add-on to VS

To assign service add-on to a VS, use the following request:

POST /virtual_machines/:virtual_machine_id/service_addons.xml

POST /virtual_machines/:virtual_machine_id/service_addons.json

XML Request Example

curl "http://onapp.test/virtual_machines/12/service_addons.xml" -d "<service_addon_id>4</service_addon_id>" -X POST \ -u user:userpass \ -H "Accept: application/xml" \ -H "Content-Type: application/xml"

JSON Request Example

curl "http://onapp.test/virtual_machines/12/service_addons.json" -d '{"service_addon_id":"4"}' -X POST \ -u user:userpass \ -H "Accept: application/json" \ -H "Content-Type: application/json"

Where:

service_addon_id – ID of the service add-on, which you want to assign to the VS
90.53 Unassign Service Add-on from VS

To unassign service add-on from the VS, use the following request:

DELETE /virtual_machines/:virtual_machine_id/service_addons/:id.xml
DELETE /virtual_machines/:virtual_machine_id/service_addons/:id.json

**XML Request Example**

```
curl "http://onapp.test/virtual_machines/12/service_addons/2.xml" -d '' -X DELETE \ -u user:userpass \ -H "Accept: application/xml" \ -H "Content-Type: application/xml"
```

**JSON Request Example**

```
curl "http://onapp.test/virtual_machines/12/service_addons/2.json" -d '' -X DELETE \ -u user:userpass \ -H "Accept: application/json" \ -H "Content-Type: application/json"
```

Where:

- **id** - ID of the service add-on, which you want to unassign from the VS

90.54 Use VS as Gateway

To make a VS function as a gateway for a network interface, use the following request:

PUT /virtual_machines/:virtual_machine_id/firewall_rules/update_defaults.xml
PUT /virtual_machines/:virtual_machine_id/firewall_rules/update_defaults.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

Where:

- **network_interface_id** - the network interface for which the VS should function as a gateway
use_as_gateway - set to '1' for the VS to function as a gateway to the network interface, otherwise, set to '0'

90.55 Virtual Server XML Config

This chapter includes API requests for managing virtual servers XML configuration.

90.55.1 Get VS XML Config

To view the XML config for a specific VS, use the following request:

GET /virtual_machines/:virtual_machine_id/xml_config.xml
GET /virtual_machines/:virtual_machine_id/xml_config.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<virtual_machine>
  <xml_config>parameters</xml_config>
  <xml_config_edited>false</xml_config_edited>
</virtual_machine>
```

Where:

xml_config - the VS XML configuration parameters, including CPU, memory and other parameters, which can differ depending on every specific VS features, OS, devices etc.
xml_config_edited - true, if XML config file was edited, otherwise, false

90.55.2 Edit VS XML Config

**IMPORTANT**

Virtual servers with modified XML configuration are not be supported by the OnApp support team.

To edit XML config for a specific VS, use the following request:
PUT /virtual_machines/:virtual_machine_id/xml_config.xml
PUT /virtual_machines/:virtual_machine_id/xml_config.json

**XML Request Example**

curl -i -X PUT -u user:userpass --url
http://onapp.test/virtual_machines/12/xml_config.xml
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
-d '<virtual_machine><xml_config>parameters</xml_config><reboot type="integer">0</reboot></virtual_machine>'

**JSON Request Example**

curl -i -X PUT -u user:userpass --url
http://onapp.test/virtual_machines/12/xml_config.json
-H 'Accept: application/json'
-H 'Content-type: application/json'
-d '{"virtual_machine": {"xml_config": "parameters","reboot": "0" }}'

Where:

- **xml_config** - edit the XML configuration parameters of the specific VS
- **reboot** - set to 1 if you want to reboot VS after editing XML configuration, otherwise, set to 0

### 90.55.3 Reset VS XML Config

To reset XML configuration to default, use the following request:

DELETE /virtual_machines/:virtual_machine_id/xml_config.xml
DELETE /virtual_machines/:virtual_machine_id/xml_config.json

**XML Request Example**

curl -i -X DELETE -u user:userpass --url
http://onapp.test/virtual_machines/12/xml_config.xml
-H 'Accept: application/xml'
-H 'Content-type: application/xml'

**JSON Request Example**

curl -i -X DELETE -u user:userpass --url
http://onapp.test/virtual_machines/12/xml_config.json
-H 'Accept: application/json'
-H 'Content-type: application/json'

### 90.56 Virtual Server Backup Resources

This chapter includes API requests for managing virtual server backup resources.

#### 90.56.1 Get List of Virtual Server Backup Resources

To get the list of virtual server backup resources, use the following request:

GET /virtual_machines/:virtual_machine_id/backups/resources.xml
GET /virtual_machines/:virtual_machine_id/backups/resources.json

XML Request Example

```bash
```

JSON Request Example

```bash
```

XML Output Example

```xml
<resources type="array">
  <resource>
    <advanced_options>
      <option_1>9</option_1>
      <option_2>8</option_2>
    </advanced_options>
    <resource_zone_id type="integer">3</resource_zone_id>
    <created_at type="dateTime">2018-04-04T12:01:31+03:00</created_at>
    <enabled type="boolean">true</enabled>
    <id type="integer">4</id>
    <label>backup_resource</label>
    <password>password</password>
    <plugin>r1soft</plugin>
    <primary_host>http://69.169.247.107:9080</primary_host>
    <secondary_host>http://69.168.257.106:9080</secondary_host>
    <updated_at type="dateTime">2018-04-04T12:17:27+03:00</updated_at>
    <username>username</username>
  </resource>
  ...</resource>
</resources>
```

Where:

- **resource** - the array of parameters for the backup resource

  - **advanced_options** - the array of advanced options for the backup resource
  - **resource_zone_id** - the ID of the backup resource zone to which the backup resource is assigned
  - **created_at** - the date when the backup resource was created in the [YYYY][MM][DD][hh][mm][ss] format
  - **enabled** - the status that indicates whether the backup resource is enabled (true) or not (false)
  - **id** - the ID of the backup resource
  - **label** - the label of the backup resource
  - **password** - the password used to connect to the third-party backup system
  - **plugin** - the label of the backup plugin
  - **primary_host** - the primary address (either hostname or IP address) used to connect to the third-party backup system
**secondary_host** - the secondary address (either hostname or IP address) used to connect to the third-party backup system

**updated_at** - the date when the backup resource was updated in the [YYYY][MM][DD][hh][mm][ss] format

**username** - the username used to connect to the third-party backup system

### Page History

v. 5.9

- updated the following API requests:
  
  - from **POST**
    
    `/virtual_servers/:virtual_server_id/backups/resources/:resource_id.xml`
    
    to **GET**
    
    `/virtual_machines/:virtual_machine_id/backups/resources.xml`
  
  - from **POST**
    
    `/virtual_servers/:virtual_server_id/backups/resources/:resource_id.json`
    
    to **GET**
    
    `/virtual_machines/:virtual_machine_id/backups/resources.json`

### 90.56.2 Add Backup Resource to Virtual Server

To add a backup resource to a virtual server, use the following request:

**POST**

`/virtual_machines/:virtual_machine_id/backups/resources/:resource_id.xml`

**POST**

`/virtual_machines/:virtual_machine_id/backups/resources/:resource_id.json`

#### XML Request Example

```
```

#### JSON Request Example

```
```

### Page History

v. 5.9

- updated the following API requests:
  
  - from **POST**
    
    `/virtual_servers/:virtual_server_id/backups/resources/:resource_id.xml`
    
    to **POST**
    
    `/virtual_machines/:virtual_machine_id/backups/resources/:resource_id.xml`
90.56.3 Remove Backup Resource from Virtual Server

To remove a backup resource from a virtual server, use the following request:

DELETE
/virtual_machines/:virtual_machine_id/backups/resources/:resource_id.xml

DELETE
/virtual_machines/:virtual_machine_id/backups/resources/:resource_id.json

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**Page History**

v. 5.9
- updated the following API requests:
  - from POST
    /virtual_servers/:virtual_server_id/backups/resources/:resource_id.json to POST
    /virtual_machines/:virtual_machine_id/backups/resources/:resource_id.json

90.57 Virtual Server Recovery Points

This chapter includes API requests for managing virtual server recovery points.

90.57.1 Create Recovery Point

To create a recovery point, use the following request:
POST /virtual_machines/:virtual_machine_id/backups/recovery_points.xml
POST /virtual_machines/:virtual_machine_id/backups/recovery_points.json

XML Request Example


JSON Request Example


Where:
resource_id - an ID of a backup resource that will be used to create a recovery point

If the request is run successfully, the 204 No Content status is returned.

90.57.2 Get List of Virtual Server Recovery Points

The get the list of virtual server recovery points, use the following request:
GET /virtual_machines/:virtual_machine_id/backups/recovery_points.xml
GET /virtual_machines/:virtual_machine_id/backups/recovery_points.json

XML Request Example


JSON Request Example


XML Output Example
<recovery_points type="array">
  <recovery_point>
    <resource_id type="integer">1</resource_id>
    <created_at type="dateTime">2018-03-15T08:05:32+00:00</created_at>
    <id type="integer">3</id>
    <size type="integer">40</size>
    <state>built</state>
    <updated_at type="dateTime">2018-03-15T08:05:32+00:00</updated_at>
    <virtual_machine_id type="integer">13</virtual_machine_id>
  </recovery_point>
  ...<recovery_point>
</recovery_points>

Where:

recovery_point - the array of recovery point parameters
resource_id - the ID of the backup resource on which the backup is created
created_at - the date when the recovery point was created in the [YYYY][MM][DD]T[hh][mm][ss] format
id - the ID of the recovery point
size - the size of the backup in bytes from which the VS can be restored
state - the status that indicates whether the backup is built
updated_at - the date when the recovery point was updated in the [YYYY][MM][DD]T[hh][mm][ss] format
virtual_machine_id - the ID of the virtual server

Page History

v. 5.9
- updated the following API requests:
  - from GET /virtual_servers/:virtual_server_id/backups/recovery_points.xml to GET /virtual_machines/:virtual_machine_id/backups/recovery_points.xml
  - from GET /virtual_servers/:virtual_server_id/backups/recovery_points.json to GET /virtual_machines/:virtual_machine_id/backups/recovery_points.json

90.57.3 Get Recovery Point Details

The get the recovery point details, use the following request:

GET /virtual_machines/:virtual_machine_id/backups/recovery_points/recovery_point_id.xml
GET /virtual_machines/:virtual_machine_id/backups/recovery_points/recovery_point_id.json

XML Request Example
Get List of File Entries for Recovery Point

The get the list of file entries for a recovery point, use the following request:

**XML Request Example**

```xml
<recovery-point>
  <id type="integer">825</id>
  <size type="integer">5368709120</size>
  <state>available</state>
  <virtual-machine-id type="integer">3024</virtual-machine-id>
  <resource-id type="integer">62</resource-id>
  <created-at type="dateTime">2018-08-08T08:00:11+03:00</created-at>
  <updated-at type="dateTime">nil="true"/></recovery-point>
```

**Where:**

- `recovery_point` - the array of recovery point parameters
- `id` - the ID of the recovery point
- `size` - the size of the backup in bytes from which the VS can be restored
- `state` - the status that indicates whether the backup is built
- `virtual_machine_id` - the ID of the virtual server
- `resource_id` - the ID of the backup resource on which the backup is created
- `created_at` - the date when the recovery point was created in the [YYYY][MM][DD][hh][mm][ss] format
- `updated_at` - the date when the recovery point was updated in the [YYYY][MM][DD][hh][mm][ss] format

JSON Request Example


XML Output Example

```xml
<file_entries type="array">
  <file_entry>
    <path>home</path>
    <file_name>home</file_name>
    <dir type="boolean">true</dir>
    <last_modified type="dateTime">2018-08-06T17:50:29+03:00</last_modified>
    <size nil="true"/>
  </file_entry>
  ...
</file_entries>
```

Where:

- **file_entries** - the array of the file entries
- **file_entry** - the array of the file entry parameters
- **path** - the path to the file entry
- **file_name** - the name of the file entry
- **dir** - true if the entry is a directory and false if the entry is a file
- **last_modified** - the date when the file entry was last modified in the [YYYY][MM][DD][T][hh][mm][ss] format
- **size** - the size of the entry in bytes that is available only for files and not for directories

90.57.5  Restore Virtual Server from Recovery Point

To restore a virtual server from a recovery point, use the following request:

POST /virtual_machines/:virtual_machine_id/backups/recovery_points/:recovery_point_id/restore.xml

POST /virtual_machines/:virtual_machine_id/backups/recovery_points/:recovery_point_id/restore.json

XML Request Example
OnApp Cloud 6.3 API Guide

**Page Histo**

v. 5.9

- updated the following API requests:
  - from POST /virtual_servers/:virtual_server_id/backups/recovery_points/:recovery_point_id/restore.xml to POST /virtual_machines/:virtual_machine_id/backups/recovery_points/:recovery_point_id/restore.xml
  - from POST /virtual_servers/:virtual_server_id/backups/recovery_points/:recovery_point_id/restore.json to POST /virtual_machines/:virtual_machine_id/backups/recovery_points/:recovery_point_id/restore.json

**90.57.6  Restore File Entries from Recovery Point**

To restore file entries from a recovery point, use the following request:

```
POST /virtual_machines/:virtual_machine_id/backups/recovery_points/:recovery_point_id/file_entries.xml
```

```
POST /virtual_machines/:virtual_machine_id/backups/recovery_points/:recovery_point_id/file_entries.json
```

**XML Request Example**

```sh
```

**JSON Request Example**

```sh
```

Where:
paths - the list of paths to the file entries starting with a slash mark (e.g. /home)

90.58 Get Virtual Server Max Memory

The request is available for the following KVM-based virtual servers:

- Virtual servers from KVM compute zones with the Ballooning release resource type and the Set max memory option enabled
- Virtual servers that are built on temples that support Hot Resize

The get a VS max memory limit, use the following request:
GET /virtual_machines/:virtual_machine_id/max_memory.xml
GET /virtual_machines/:virtual_machine_id/max_memory.json

XML Request Example


JSON Request Example


XML Output Example

```xml
<virtual_machine>
  <max_memory_override type="boolean">true</max_memory_override>
  <preset_max_memory type="integer">2439</preset_max_memory>
</virtual_machine>
```

Where:
max_memory_override - true if the max memory limit is enabled, otherwise, false
preset_max_memory - the maximum amount of RAM in MB allocated to the VS
90.59 Edit Virtual Server Max Memory

The request is available for the following KVM-based virtual servers:
- Virtual servers from KVM compute zones with the Ballooning release resource type and the Set max memory option enabled
- Virtual servers that are built on templates that support Hot Resize

The edit a VS max memory limit, use the following request:
PUT /virtual_machines/:virtual_machine_id/max_memory.xml
PUT /virtual_machines/:virtual_machine_id/max_memory.json

XML Request Example

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/virtual_machines/12/max_memory.xml -d '<virtual_machine><max_memory_override type="boolean">true</max_memory_override><preset_max_memory type="integer">200</preset_max_memory></virtual_machine>'
```

JSON Request Example

```bash
```

Where:
- max_memory_override - true to enable the max memory limit, otherwise, false
- preset_max_memory - the maximum amount of RAM in MB allocated to the VS

90.60 Enable Virsh Console

To enable Virsh console for a virtual server, use the following request:
POST /virtual_machines/:id/virsh_console.xml
POST /virtual_machines/:id/virsh_console.json

XML Request Example

```bash
```

JSON Request Example
The **200 OK** status is returned when the request is completed successfully.

### 90.61 Disable Virsh Console

To disable Virsh console for a virtual server, use the following request:

```
DELETE /virtual_machines/:id/virsh_console.xml
DELETE /virtual_machines/:id/virsh_console.json
```

**XML Request Example**
```
```

**JSON Request Example**
```
```

The **204 No content** status is returned when the request is completed successfully.

### 90.62 Add/Edit OVA VS License

This functionality is available only for virtual servers created from the OVAs with the Network Appliance operating system.

To add or edit OVA VS license, use the following request:

```
PUT /virtual_machines/:vm_identifier/network_appliance_license.xml
PUT /virtual_machines/:vm_identifier/network_appliance_license.json
```

**XML Request Example**
```
```

**JSON Request Example**
```
```

Where:

- **network_appliance_license** - label of the network appliance license
- **file_url** - the URL to your network appliance config file. If this parameter is empty, it will be ignored and only the **network_appliance_license** parameter will be applied.

90.63 Add/Edit OVA VS Config

This functionality is available only for virtual servers created from the OVAs with the Network Appliance operating system.

To add new OVA VS config or edit the existing one, use the following request:

PUT /virtual_machines/:vm_identifier/network_appliance_config.xml
PUT /virtual_machines/:vm_identifier/network_appliance_config.json

**XML Request Example**


**JSON Request Example**


Where:

- **network_appliance_config** - label of the network appliance config
- **file_url** - the URL path to your network appliance config file. If this parameter is empty, it will be ignored and only the **network_appliance_license** parameter will be applied.
91 Virtual Routers

This chapter describes how to manage virtual routers.

91.1 Get List of Virtual Routers

To get the list of virtual routers, use the following request:

GET /virtual_routers.xml.
GET /virtual_routers.json

XML Request Example

```bash
```

JSON Request Example

```bash
```
<virtual_routers type="array">
  <virtual_router>
    <id type="integer">23</id>
    <hypervisor_id type="integer">40</hypervisor_id>
    <template_id type="integer">2</template_id>
    <identifier>qcadpfahklfzbh</identifier>
    <hostname>testVR</hostname>
    <memory type="integer">128</memory>
    <cpus type="integer">1</cpus>
    <cpu_shares type="integer">1</cpu_shares>
    <created_at type="dateTime">2018-12-10T15:59:02:00</created_at>
    <updated_at type="dateTime">2018-12-10T15:46:27+02:00</updated_at>
    <built type="boolean">true</built>
    <locked type="boolean">false</locked>
    <booted type="boolean">true</booted>
    <xen_id type="integer">3</xen_id>
    <remote_access_password>uM14pTxy2xyM</remote_access_password>
    <local_remote_access_port type="integer">5900</local_remote_access_port>
    <label>testVR</label>
    <recovery_mode nil="true"/>
    <user_id type="integer">1</user_id>
    <operating_system>linux</operating_system>
    <operating_system_distro>ubuntu</operating_system_distro>
    <allowed_swap type="boolean">true</allowed_swap>
    <template_label>Debian 5.0 x86</template_label>
    <min_disk_size type="integer">5</min_disk_size>
    <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
    <note nil="true"/>
    <admin_note nil="true"/>
    <suspended type="boolean">false</suspended>
    <enable_autoscale type="boolean">false</enable_autoscale>
    <add_to_marketplace nil="true"/>
    <state>delivered</state>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <edge_server_type nil="true"/>
    <storage_server_type nil="true"/>
    <firewall_notrack type="boolean">false</firewall_notrack>
    <service_password nil="true"/>
    <preferred_hVR type="array"/>
    <local_remote_access_ip_address>10.0.51.131</local_remote_access_ip_address>
    <cpu_units type="integer">10</cpu_units>
    <cpu_sockets nil="true"/>
    <iso_id nil="true"/>
    <cores_per_socket type="array">0</cores_per_socket
    <instance_package_id nil="true"/>
    <hot_add_cpu nil="true"/>
    <hot_add_memory nil="true"/>
    <time_zone nil="true"/>
    <autoscale_service nil="true"/>
    <cdboot type="boolean">false</cdboot>
    <draas_mode type="integer">0</draas_mode>
    <vapp_id nil="true"/>
    <vmware_tools nil="true"/>
    <vcenter_moref nil="true"/>
    <template_version>1.2</template_version>
    <openstack_id nil="true"/>
    <domain>localdomain</domain>
    <vcenter_reserved_memory type="integer">0</vcenter_reserved_memory>
    <deleted_at nil="true"/>
  </virtual_router>
</virtual_routers>
Where:

**id** - the virtual router ID  
**hypervisor_id** - the ID of a compute resource the IP address is associated with  
**template_id** - the ID of the template the VR is based on  
**identifier** - the VR identifier  
**hostname** - the name of your host  
**memory** - the RAM size allocated to this virtual router  
**cpus** - the number of allocated CPU cores  
**cpu_shares** - CPU priority in percents  
**created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format  
**updated_at** - the date when the VR was updated in the [YYYY][MM][DD][hh][mm][ss]Z format  
**built** - true if the VR is built, otherwise false  
**locked** - true if the VR is locked; otherwise false
booted - true if the VR is running, otherwise false

xen_id - the VR ID set by the virtualization engine

remote_access_password - the password for the remote access

local_remote_access_port - the port ID used for console access

cpu_units - the amount of CPU units per core if the CPU priority is replaced with CPU units in user bucket.

cpu_sockets - the amount of CPU sockets per core. This parameter can be set for KVM compute resources only by those users who have Enable CPU topology permission granted

label - the VR label

recovery_mode - true if recovery mode allowed. Otherwise false

user_id - the ID of a user assigned to this VR

operating_system - operating system used by the VR

operating_system_distro - the distribution of the OS from which this VR is built

allowed_swap - true if swap disk is allowed (depends on the template the VR is based on); otherwise false

template_label - the name of the template from which this VR is built

min_disk_size - the minimum disk size required to build a VR from a specified template

allowed_hot_migrate - true if the template, on which the VR is based, supports hot migration; otherwise false

note - an optional reminder for this VR made by a user account

admin_note - an optional note of the administrator

suspended - true if VR is suspended, otherwise false

strict_virtual_machine_id - the ID of a virtual server that will never reside on the same compute resource with this VR

enable_autoscale - true if autoscaling is allowed for this VR

add_to_marketplace - empty for VRs; used for edge servers only

state – parameter reserved for future use

initial_root_password_encrypted - true, if the root password is encrypted, otherwise false.

edge_server_type - true if this is the edge server

storage_server_type - true if this is a storage server
**firewall_notrack** - true if the NOTRACK rule is set in `iptables`

**service_password** - service account password

**preferred_hVR** - the array of preferable compute resources based on compute zone that meet some VR configuration settings

**local_remote_access_ip_address** - IP address used for remote access

**iso_id** - the ID of the ISO the VR is based on

**cores_per_socket** - the amount of cores per socket

**instance_package_id** - ID of the instance package

**hot_add_cpu** - true, if the CPU parameter can be changed without rebooting the VR, otherwise false

**hot_add_memory** - true, if the memory parameter can be changed without rebooting the VR, otherwise false

**time_zone** - the time zone set for the VR. This parameter is applicable only to Windows KVM and XEN virtual servers.

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows VR manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a VR after starting it if time synchronization is not completed for some reason.

**draas_mode** - true if DRaaS is enabled for the Cloud; otherwise false

**domain** - specify the domain for this VR

**vcenter_reserved_memory** - amount of RAM assigned to the vCenter VS

**acceleration_allowed** - true if acceleration is enabled for the virtual server; otherwise false.

**ip_addresses** - an array of IP addresses assigned to this VR and their details:

- **id** - the ID of the IP address
- **address** - IP address
- **broadcast** - broadcast address
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **customer_network_id** - the ID of the customer VLAN the IP address belongs to
- **disallowed_primary** - true if not allowed to be used as primary, otherwise false
- **gateway** - gateway address
- **id** - the ID of the IP address
- **ip_address_pool_id** - ID of the IP address pool the IP address is associated with
- **pxe** - true, if this address can be used for cloudbooting a compute resource
- **updated_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **user_id** - the ID of the user this IP address is assigned to
- **free** - true if free, otherwise false
- **netmask** - netmask for the IP address

**monthly_bandwidth_used** - VR monthly bandwidth in KB
**total_disk_size** - the total disk size in GB of all disks assigned to VR

**support_incremental_backups** - 1, if the virtual server supports incremental backups, and 0 if it does not

cpu_priority - this is a new parameter reserved for further use; currently will have the same value as cpu_shares

**built_from_iso** - true if the VR is built from ISO; otherwise false

**built_from_ova** - true if the VR is built from OVA; otherwise false

**acceleration** - true if acceleration is enabled for the VR; otherwise false

**acceleration_status** - the status of acceleration: active or inactive

**hypervisor_type** - the type of the compute resource the VR is built on. Currently, only KVM type is available.

**initial_root_password** - the VR root password

**vip** - true if the VR has VIP status (gives migration priority)

**price_per_hour** - router's price per hour

**price_per_hour_powered_off** - price per hour when server is powered off

---

### 91.2 Get Virtual Router Details

To get the details of the particular virtual router, use the following request:

GET /virtual_routers/:virtual_router_id.xml

GET /virtual_routers/:virtual_router_id.json

**XML Request Example**

```
```

**JSON Request Example**

```
```

**XML Output Example**
<virtual_router>
  <id type="integer">45</id>
  <hypervisor_id type="integer">4</hypervisor_id>
  <template_id type="integer">4</template_id>
  <identifier>urhzttqfaxnmo</identifier>
  <hostname>VirtualRouter</hostname>
  <memory type="integer">384</memory>
  <cpus type="integer">1</cpus>
  <cpu_shares type="integer">1</cpu_shares>
  <created_at type="dateTime">2018-12-11T11:41:58+02:00</created_at>
  <updated_at type="dateTime">2018-12-11T17:29:00+02:00</updated_at>
  <built type="boolean">true</built>
  <locked type="boolean">false</locked>
  <booted type="boolean">true</booted>
  <xen_id type="integer">49</xen_id>
  <remote_access_password>BSNrg9mXMh8ec</remote_access_password>
  <local_remote_access_port type="integer">5902</local_remote_access_port>
  <label>VirtualRouter</label>
  <recovery_mode type="boolean">false</recovery_mode>
  <user_id type="integer">8</user_id>
  <operating_system>linux</operating_system>
  <operating_system_distro>rhel</operating_system_distro>
  <allowed_swap type="boolean">true</allowed_swap>
  <template_label>CentOS 7.5 x64</template_label>
  <min_disk_size type="integer">5</min_disk_size>
  <allowed_hot_migrate type="boolean">true</allowed_hot_migrate>
  <note nil="true"/>
  <admin_note nil="true"/>
  <suspended type="boolean">false</suspended>
  <enable_autoscale type="boolean">false</enable_autoscale>
  <add_to_marketplace nil="true"/>
  <state>delivered</state>
  <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
  <storage_server_type nil="true"/>
  <firewall_notrack type="boolean">false</firewall_notrack>
  <service_password nil="true"/>
  <preferred_hvs type="array"/>
  <local_remote_access_ip_address>10.0.24.32</local_remote_access_ip_address>
    <cpu_units type="integer">10</cpu_units>
    <cpu_sockets nil="true"/>
    <draas_keys type="array"/>
    <iso_id nil="true"/>
    <cores_per_socket type="integer">0</cores_per_socket>
    <instance_package_id nil="true"/>
    <hot_add_cpu nil="true"/>
    <hot_add_memory nil="true"/>
    <time_zone nil="true"/>
    <autoscale_service>0</autoscale_service>
    <cdboot type="boolean">false</cdboot>
    <draas_mode type="integer">0</draas_mode>
    <vapp_id nil="true"/>
    <vmware_tools nil="true"/>
    <vcener_moref nil="true"/>
    <template_version>1.1</template_version>
    <openstack_id nil="true"/>
    <domain>localdomain</domain>
    <vcener_reserved_memory type="integer">0</vcener_reserved_memory>
    <deleted_at nil="true"/>
  </properties>
</virtual_router>
<ip_addresses type="array">
  <networking_ip_address>
    <id type="integer">609</id>
    <address>69.168.246.51</address>
    <broadcast>69.168.246.253</broadcast>
    <network_address>69.168.246.0</network_address>
    <gateway>69.168.237.1</gateway>
    <created_at type="dateTime">2018-12-11T17:11:51+02:00</created_at>
    <updated_at type="dateTime">2018-12-11T17:11:51+02:00</updated_at>
    <user_id nil="true"/>
    <pxe type="boolean">false</pxe>
    <hypervisor_id nil="true"/>
    <ip_range_id type="integer">5</ip_range_id>
    <free type="boolean">false</free>
    <netmask>255.255.255.0</netmask>
  </networking_ip_address>
</ip_addresses>

<monthly_bandwidth_used type="decimal">0.068296996</monthly_bandwidth_used>
<total_disk_size type="integer">6</total_disk_size>

Where:

- **id** – the virtual router ID
- **hypervisor_id** – the ID of a compute resource the IP address is associated with
- **template_id** – the ID of the template the virtual router (VR) is based on
- **identifier** – the VR identifier
- **hostname** – the name of your host
- **memory** – the RAM size allocated to this virtual router
- **cpus** – the number of allocated CPU cores
- **cpu_shares** – CPU priority in percents
- **created_at** – the date in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **updated_at** – the date when the VR was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- **built** – true if the VR is built, otherwise false
- **locked** – true if the VR is locked; otherwise false
- **booted** – true if the VR is running, otherwise false
- **xen_id** – the VR ID set by the virtualization engine
- **remote_access_password** – the password for the remote access
- **local_remote_access_port** – the port ID used for console access
- **label** – the VR label
recovery_mode – true if recovery mode allowed. Otherwise false
user_id – the ID of a user assigned to this VR
operating_system – operating system used by the VR
operating_system_distro – the distribution of the OS from which this VR is built
allowed_swap – true if swap disk is allowed (depends on the template the VR is based on); otherwise false
template_label – the name of the template from which this VR is built
min_disk_size – the minimum disk size required to build a VR from a specified template
allowed_hot_migrate – true if the template, on which the VR is based, supports hot migration; otherwise false
note – an optional reminder for this VR made by a user account
admin_note – an optional note of the administrator
suspended – true if VR is suspended, otherwise false
strict_virtual_machine_id – the ID of a virtual server that will never reside on the same compute resource with this VR
enable_autoscale – true if autoscaling is allowed for this VR
add_to_marketplace – empty for VRs; used for edge servers only
state – parameter reserved for future use
initial_root_password_encrypted – true, if the root password is encrypted, otherwise false.
edge_server_type – true if this is the edge server
storage_server_type – true if this is a storage server
firewall_notrack – true if the NOTRACK rule is set in iptables
service_password – service account password
preferred_hvs – the array of preferable compute resources based on compute zone that meet some VR configuration settings
local_remote_access_ip_address – IP address used for remote access
cpu_units – the amount of CPU units per core if the CPU priority is replaced with CPU units in user billing plan.
cpu_sockets – the amount of CPU sockets per core.
isod – the ID of the ISO the VR is based on
cores_per_socket – the amount of cores per socket
instance_package_id – ID of the instance package
hot_add_cpu – true, if the CPU parameter can be changed without rebooting the VR, otherwise false
hot_add_memory – true, if the memory parameter can be changed without rebooting the VR, otherwise false
time_zone – the time zone set for the VR. This parameter is applicable only to Windows KVM and XEN virtual servers.

Currently, the time zone is set at the Compute resource side only. Therefore, users need to set the target time zone inside a Windows VR manually. Setting correct time zone at the Compute resource side helps to keep correct time inside a VR after starting it if time synchronization is not completed for some reason.
draas_mode – true if DRaaS is enabled for the Cloud; otherwise false
domain – specify the domain for this VR

vcenter_reserved_memory – amount of RAM assigned to the vCenter VS

acceleration_allowed – true if acceleration is enabled for the virtual server; otherwise false.

ip_addresses – an array of IP addresses assigned to this VR and their details:
- **id** – the ID of the IP address
- **address** – IP address
- **broadcast** – broadcast address
- **network_address** – the network address of the IP net
- **gateway** – gateway address
- **created_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **user_id** – the ID of the user this IP address is assigned to
- **pxe** – true, if this address can be used for cloudbooting a compute resource
- **hypervisor_id** – the ID of a compute resource the IP address is associated with
- **ip_range_id** – the ID of the IP range from which the IP address should be assigned
- **free** – true if free, otherwise false
- **netmask** – netmask for the IP address

monthly_bandwidth_used – VR monthly bandwidth in KB

total_disk_size – the total disk size in GB of all disks assigned to VR

support_incremental_backups – 1, if the virtual server supports incremental backups, and 0 if it does not

cpu_priority – this is a new parameter reserved for further use; currently will have the same value as cpu_shares

built_from_iso – true if the VR is built from ISO; otherwise false

built_from_ova – true if the VR is built from OVA; otherwise false

acceleration – true if acceleration is enabled for the VR; otherwise false

acceleration_status – the status of acceleration: active or inactive

hypervisor_type – the type of the compute resource the VR is built on. Currently, only KVM type is available.

initial_root_password – the VR root password

vip – true if the VR has VIP status (gives migration priority)

price_per_hour – router's price per hour

price_per_hour_powered_off – price per hour when server is powered off

---

### 91.3 Convert Virtual Server to Virtual Router

To convert a virtual server to a virtual router, use the following request:

**PUT**

/virtual_machines/:virtual_machine_id/convert_to_virtual_router.xml
PUT
/virtual_machines/:virtual_machine_id/convert_to_virtual_router.json

XML Request Example

curl -i -X PUT -u user:userpass --url
http://onapp.test/virtual_machines/65/convert_to_virtual_router.xml -H
'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X PUT -u user:userpass --url
http://onapp.test/virtual_machines/65/convert_to_virtual_router.json -H
'Accept: application/json' -H 'Content-type: application/json'

91.4 Get List of IP Nets Assigned to Virtual Router

To get the list of IP nets assigned to a virtual router, use the following request:
GET /virtual_routers/:virtual_router_id/ip_nets.xml
GET /virtual_routers/:virtual_router_id/ip_nets.json

XML Request Example

curl -i -X GET -u user:userpass --url

JSON Request Example

curl -i -X GET -u user:userpass --url

XML Output Example

```xml
<ip_nets type="array">
  <ip_net>
    <id type="integer">87</id>
    <network_id type="integer">92</network_id>
    <network_mask>24</network_mask>
    <ipv4 type="boolean">true</ipv4>
    <label>sdn-ip-net-3</label>
    <gateway_outside_ip_net>false</gateway_outside_ip_net>
    <network_address>10.10.16.0</network_address>
    <default_gateway>10.10.16.1</default_gateway>
  </ip_net>
  ...</ip_net>
</ip_nets>
```

Where:
id - the ID of the IP net

network_id - the ID of the network

network_mask - the network mask

ipv4 - whether this is an IPv4 or an IPv6 IP net: `true` for IPv4 networks and `false` for IPv6 networks

label - the name of the IP net

gateway_outside_ip_net - true if the gateway is outside from the IP net

network_address - the network address of the IP net

default_gateway - external gateway IP address

91.5 Get List of Attachable IP Nets

To get the list of the IP nets that can be attached to a virtual router, use the following request:

GET /virtual_routers/:virtual_router_id/attachable_ip_nets.xml
GET /virtual_routers/:virtual_router_id/attachable_ip_nets.json

XML Request Example

curl -i -X GET -u user:userpass --url
http://onapp.test/virtual_routers/xxahjttyelztxr/attachable_ip_nets.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request Example

curl -i -X GET -u user:userpass --url
http://onapp.test/virtual_routers/xxahjttyelztxr/attachable_ip_nets.json
-H 'Accept: application/json' -H 'Content-type: application/json'

XML Output Example

```xml
<ip_nets type="array">
  <ip_net>
    <id type="integer">46</id>
    <network_id type="integer">4</network_id>
    <network_mask>24</network_mask>
    <ipv4 type="boolean">true</ipv4>
    <label>fakeNetwork6.6.0.0</label>
    <gateway_outside_ip_net>false</gateway_outside_ip_net>
    <network_address>168432896</network_address>
    <default_gateway>168432897</default_gateway>
  </ip_net>
  ...</ip_net>
</ip_nets>
```

Where:
**91.6 Assign IP Net to Virtual Router**

To assign an IP net to a virtual router, use the following request:

```plaintext
POST /virtual_routers/:virtual_router_id/assign_ip_net.xml
POST /virtual_routers/:virtual_router_id/assign_ip_net.json
```

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

**Where:**

- `ip_net_id` - the ID of the IP net you want to assign to the virtual router

**91.7 Unassign IP Net from Virtual Router**

To unassign IP net from a virtual router, use the following request:

```plaintext
DELETE /virtual_routers/:virtual_router_id/unassign_ip_net.xml
DELETE /virtual_routers/:virtual_router_id/unassign_ip_net.json
```

**XML Request Example**

```bash
```

**JSON Request Example**

```bash
```

Where:

*ip_net_id* - the ID of the IP net you want to unassign from the virtual router
92 Whitelist IPs

Whitelist IP addresses are IP addresses from which a particular user can access the OnApp control panel. If whitelisted IP addresses are specified for a particular user, the user can only access CP from that defined IP addresses.

92.1 Get List of Whitelist IPs

To get the list of IPs entered to the list, use the following request:

GET /users/:user_id/user_white_lists.xml
GET /users/:user_id/user_white_lists.json

XML Request Example

```
```

JSON Request Example

```
```

XML Output Example

```
<user_white_lists>
    <user_white_list>
        <created_at>2011-04-21T15:38:14+03:00</created_at>
        <description>My IP</description>
        <id>2</id>
        <ip>192.168.112.1</ip>
        <updated_at>2011-04-21T15:38:14+03:00</updated_at>
        <user_id>8</user_id>
    </user_white_list>
    ...
    <user_white_list></user_white_list>
    ...
    </user_white_lists>
```

Where:

- `created_at` - the date when this record in DB was created
- `description` - an optional description
- `id` - the record ID
- `ip` - the IP from which this user can log in to CP
- `updated_at` - the date when this record in DB was updated
- `user_id` - the ID of a user for whom this whitelist was created
92.2 Get Whitelist IP Details

To get details for a particular whitelist, use the following request:

GET /users/:user_id/user_white_lists/:id.xml
GET /users/:user_id/user_white_lists/:id.json

**XML Request Example**


**JSON Request Example**


**XML Output Example**

```xml
<user_white_list>
  <created_at>2011-04-21T15:38:14+03:00</created_at>
  <description>My IP</description>
  <id>2</id>
  <ip>192.168.112.1</ip>
  <updated_at>2011-04-21T15:38:14+03:00</updated_at>
  <user_id>8</user_id>
</user_white_list>
```

Where:

- **created_at** - the date when this record in DB was created
- **description** - an optional description
- **id** - the record ID
- **ip** - the IP from which this user can log in to CP
- **updated_at** - the date when this record in DB was updated
- **user_id** - the ID of a user for whom this whitelist was created

92.3 Add Whitelisted IP

To add an IP to the list of whitelisted IPs:

POST /users/:user_id/user_white_lists.xml
POST /users/:user_id/user_white_lists.json

**XML Request Example**

```xml
<user_white_list>
  <created_at>2011-04-21T15:38:14+03:00</created_at>
  <description>My IP</description>
  <id>2</id>
  <ip>192.168.112.1</ip>
  <updated_at>2011-04-21T15:38:14+03:00</updated_at>
  <user_id>8</user_id>
</user_white_list>
```
OnApp Cloud 6.3 API Guide

```bash
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '
  <user_white_list><ip>127.0.0.111</ip></user_white_list>'
  http://onapp.test/users/9/user_white_lists.xml
```

**JSON Request Example**

```bash
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -d
  '{"user_white_list":{"ip":"109.123.105.178","description":"qsas"}}' -u
  user:userpass http://onapp.test/users/:user_id/user_white_lists.json
```

Where:

*ip* - IP address, from which a user can login to the Control panel

### 92.4 Edit Whitelisted IP

To edit a whitelisted IP, use the following request:

PUT /users/:user_id/user_white_lists/:id.xml
PUT /users/:user_id/user_white_lists/:id.json

**XML Request Example**

```bash
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<user_white_list><ip>127.0.0.123</ip></user_white_list>'
  *http://onapp.test/users/8/user_white_lists/9.xml
```

**JSON Request Example**

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -d
  '{"user_white_list":{"ip":"109.123.105.129","description":"udr"}}' -u
  user:userpass http://onapp.test/users/8/user_white_lists/9.json
```

### 92.5 Delete Whitelisted IP

To delete a whitelisted IP, use the following request:

DELETE /users/:user_id/user_white_lists/:id.xml
DELETE /users/:user_id/user_white_lists/:id.json

**XML Request Example**

```bash
curl -X DELETE -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass
  http://onapp.test/users/8/user_white_lists/9.xml
```
JSON Request Example

curl -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass
http://onapp.test/users/8/user_white_lists/9.json
93 Zabbix Server

Zabbix is used for autoscaling of newly created VSs.

We strongly do not recommend installing Zabbix on the Control Panel server. You can use a separate server or a VS (if your network allows it) as the Zabbix server.

93.1 Zabbix Server Setup

To set up a Zabbix server, use the following request:

- POST /sysadmin_tools/infrastructure/zabbix_setup/deploy.xml
- POST /sysadmin_tools/infrastructure/zabbix_setup/deploy.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/sysadmin_tools/infrastructure/zabbix_setup/deploy.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d
'<zabbix_setup><ip_address>65.178.209.17</ip_address><server_os>centos6</server_os></zabbix_setup>'
```

**JSON Request Example**

```
curl -i -X POST -u 'user:userpass' --url
http://onapp.test/sysadmin_tools/infrastructure/zabbix_setup/deploy.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d
'{"zabbix_setup": {"ip_address": "65.178.209.17", "server_os": "centos7"}}'
```

Where:
- **ip_address** - IP address of the Zabbix server
- **server_os** - operating system of the Zabbix server

If the request is run successfully, the 204 No Content status is returned.

93.2 Reconfigure Existing Zabbix Server

To reconfigure an existing Zabbix server, use the following request:

- POST /sysadmin_tools/infrastructure/zabbix_setup/configure.xml
- POST /sysadmin_tools/infrastructure/zabbix_setup/configure.json

**XML Request Example**

```
curl -i -X POST -u user:userpass --url
http://onapp.test/sysadmin_tools/infrastructure/zabbix_setup/configure.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

```
curl -i -X POST -u user:userpass --url
http://onapp.test/sysadmin_tools/infrastructure/zabbix_setup/configure.json
-H 'Accept: application/json' -H 'Content-type: application/json'
```
JSON Request Example

curl -i -X POST -u 'user:userpass' --url

If the request is run successfully, the 204 No Content status is returned.