# TABLE OF CONTENTS

1. API Authentication ................................................. 47
2. HTTP Methods .......................................................... 48
3. HTTP response codes .................................................. 49
4. Formatting and naming conventions ............................... 50
5. FAQ ............................................................................ 51
6. OnApp 5.3 .................................................................. 52
7. Add Zabbix Server ...................................................... 54
8. Alerts ........................................................................... 55
9. Application Servers ..................................................... 56
   9.1 Create Application Server .......................................... 56
   9.1.1 Page History .......................................................... 61
10. Apps for Application Servers ......................................... 62
   10.1 Get List of All Installed Applications .......................... 62
   10.2 Get List of All Available for Installation Applications .... 63
   10.3 Get Application Attributes ....................................... 64
   10.4 Install Application ................................................... 65
   10.5 Back Up Application ................................................ 65
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.6</td>
<td>Delete Application</td>
<td>66</td>
</tr>
<tr>
<td>10.7</td>
<td>Get List of All Application Backups</td>
<td>67</td>
</tr>
<tr>
<td>10.8</td>
<td>Restore Application Backup</td>
<td>68</td>
</tr>
<tr>
<td>10.9</td>
<td>Remove Application Backup</td>
<td>69</td>
</tr>
<tr>
<td>10.10</td>
<td>System Applications</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>10.10.1 Get List of System Applications</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>10.10.2 Install System Application</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>10.10.3 Switch PHP Version</td>
<td>71</td>
</tr>
<tr>
<td></td>
<td>10.10.4 Uninstall System Application</td>
<td>71</td>
</tr>
<tr>
<td>10.11</td>
<td>Domains</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td>10.11.1 Get List of Domains</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td>10.11.2 Create Domain</td>
<td>73</td>
</tr>
<tr>
<td></td>
<td>10.11.3 Delete Domain</td>
<td>75</td>
</tr>
<tr>
<td>10.12</td>
<td>FTP Users</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>10.12.1 Get List of FTP Users</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>10.12.2 Create FTP User</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>10.12.3 Delete FTP User</td>
<td>77</td>
</tr>
<tr>
<td>10.13</td>
<td>Databases</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>10.13.1 Get List of Databases</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>10.13.2 Create Database</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>10.13.3 Delete Database</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>10.13.4 Get List of Database Users</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>10.13.5 Get List of Users Assigned to Database</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>10.13.6 Create Database User</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>10.13.7 Assign User to Database</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>10.13.8 Update Database User Privileges</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>10.13.9 Change Database User Password</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>10.13.10 Unassign User from Database</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>10.13.11 Delete Database User</td>
<td>85</td>
</tr>
<tr>
<td>10.14</td>
<td>Email Accounts</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>10.14.1 Get List of Email Accounts</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td>10.14.2 Get List of Email Accounts for Specific Domain</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td>10.14.3 Create Email Account</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>10.14.4 Delete Email Account for Default Domain</td>
<td>88</td>
</tr>
<tr>
<td></td>
<td>10.14.5 Delete Email Account for Specific Domain</td>
<td>88</td>
</tr>
<tr>
<td>10.15</td>
<td>Services</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>10.15.1 Get List of Services</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>10.15.2 Start/Stop/Restart Services</td>
<td>90</td>
</tr>
</tbody>
</table>
13.7 Create Backups for All Disks ................................................................. 126
13.8 Convert Backup to Template ................................................................. 127
13.9 Delete Backup ...................................................................................... 127
13.10 Restore Backup .................................................................................. 128
13.11 Add/Edit Backup Note ......................................................................... 128

14 Backup Servers ................................................................. 129
14.1 Get List of Backup Servers .................................................................. 129
14.2 Get Backup Server Details .................................................................. 130
14.3 Add Backup Server ............................................................................ 132
14.4 Edit Backup Server ............................................................................ 133
14.5 Delete Backup Server ......................................................................... 134
14.6 Search Backups ................................................................................... 134
14.7 Create Cloud Boot Backup Server ...................................................... 137

15 Backup Server Zones ................................................................. 139
15.1 Get List of Backup Server Zones ....................................................... 139
15.1.1 Page history .................................................................................. 139
15.2 Get Backup Server Zone Details ....................................................... 139
15.2.1 Page history ................................................................................ 140
15.3 Add Backup Server Zone .................................................................... 140
15.3.1 Page history ................................................................................ 141
15.4 Edit Backup Server Zone .................................................................... 141
15.4.1 Page history ................................................................................ 141
15.5 Delete Backup Server Zone .................................................................. 142
15.6 Get List of Servers Assigned to Backup Server Zone ......................... 142
15.7 Assign Backup Server to Backup Server Zone .................................... 143
15.8 Unassign Backup Server from Backup Server Zone ............................ 144
16 Baremetal Servers .......................................................... 145
   16.1 Get List of Baremetal Servers ........................................ 145
   16.2 Get Baremetal Server Details ........................................... 147
   16.3 Create Baremetal Server .................................................. 150
   16.4 Delete Baremetal Server ................................................... 151
   16.5 Add/Edit Admin/User Note for Baremetal Server ...................... 151
   16.6 Enable Recovery Mode for Baremetal Server ........................... 152
   16.7 Disable Recovery Mode for Baremetal Server .......................... 153

17 Billing Plans ................................................................. 154
   17.1 Get the List of Billing Plans ............................................ 157
      17.1.1 Page history ......................................................... 158
   17.2 Get Billing Plan Details .................................................. 159
      17.2.1 Page history ........................................................ 160
   17.3 Add Billing Plan ............................................................ 161
      17.3.1 Page history ........................................................ 161
   17.4 Edit Billing Plan ............................................................ 161
      17.4.1 Page history ........................................................ 162
   17.5 Get List of Base Resources .............................................. 162
      17.5.1 Page history ........................................................ 166
   17.6 Get Base Resources Details .............................................. 167
      17.6.1 Where: ............................................................... 168
      17.6.2 Page History ....................................................... 177
   17.7 Get Master Template/Master Bucket Details ............................ 179
      17.7.1 Where: ............................................................... 180
      17.7.2 Where: ............................................................... 182
      17.7.3 Where: ............................................................... 184
   17.8 Add Base Resources to Billing Plan ..................................... 185
      17.8.1 Add User VS Limits .................................................. 185
      17.8.2 Add Limits for Template Store ...................................... 187
      17.8.3 Add Limits for Recipe Groups ...................................... 188
      17.8.4 Add Limits for Compute Zones ...................................... 188
      17.8.5 Add Limits for Data Store Zones .................................... 192
OnApp 5.3 API Guide

17.8.6 Add Limits for Network Zones ................................................................. 195
17.8.7 Add Limits for Edge Groups ................................................................. 197
17.8.8 Add Limits for Backup Server Zones ..................................................... 198
17.8.9 Add Limits for minIOPS ........................................................................ 200
17.8.10 Add Limits for Instance Packages ......................................................... 201
17.8.11 Add Limits for Service Add-on Groups .................................................. 202

17.9 Edit Base Resources .................................................................................... 203
17.9.1 Page history ............................................................................................ 207

17.10 Edit Limits & Pricing for Compute Zones .................................................. 207

17.11 Edit Limits & Pricing for Data Store Zones ............................................... 211
17.11.1 Page history .......................................................................................... 213

17.12 Edit Limits & Pricing for Network Zones .................................................. 213
17.12.1 Page history .......................................................................................... 215

17.13 Edit Limits & Pricing for Backup Server Zones ......................................... 215
17.13.1 Page history .......................................................................................... 218

17.14 Delete Base Resource From Billing Plan .................................................. 219

17.15 Add Base Resource to Master Bucket ...................................................... 220
17.15.1 Page history .......................................................................................... 220

17.16 Remove Base Resource from Master Bucket .......................................... 220
17.16.1 Page history .......................................................................................... 221

17.17 Add Base Resource to Master Template ................................................. 221
17.17.1 Page history .......................................................................................... 222

17.18 Remove Base Resource From Master Template ...................................... 222
17.18.1 Page history .......................................................................................... 223

17.19 Delete Billing Plan ..................................................................................... 223
17.19.1 Page history .......................................................................................... 223

18 Blueprints ........................................................................................................ 224

18.1 Get List of Blueprints ................................................................................. 224
18.2 Get Blueprint Details ................................................................................. 225
18.3 Add Blueprint .............................................................................................. 226
18.4 Delete Blueprint ........................................................................................... 227
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>Blueprint Template Groups</td>
<td></td>
</tr>
<tr>
<td>19.1</td>
<td>Get List of Blueprint Template Groups</td>
<td>228</td>
</tr>
<tr>
<td>19.2</td>
<td>Get Blueprint Template Group Details</td>
<td>230</td>
</tr>
<tr>
<td>19.3</td>
<td>Add Blueprint Template Group</td>
<td>231</td>
</tr>
<tr>
<td>19.4</td>
<td>Edit Blueprint Template Group</td>
<td>232</td>
</tr>
<tr>
<td>19.5</td>
<td>Delete Blueprint Template Group</td>
<td>233</td>
</tr>
<tr>
<td>19.6</td>
<td>Add Child Blueprint Template Group</td>
<td>234</td>
</tr>
<tr>
<td>19.7</td>
<td>Get List of Blueprint Templates Attached to</td>
<td>234</td>
</tr>
<tr>
<td></td>
<td>Blueprint Template Group</td>
<td></td>
</tr>
<tr>
<td>19.8</td>
<td>Attach Blueprint Template to Group</td>
<td>236</td>
</tr>
<tr>
<td>19.9</td>
<td>Remove Blueprint Template from Blueprint Template</td>
<td>236</td>
</tr>
<tr>
<td>20</td>
<td>Blueprint Templates</td>
<td></td>
</tr>
<tr>
<td>20.1</td>
<td>Get List of Blueprint Templates</td>
<td>238</td>
</tr>
<tr>
<td>20.2</td>
<td>Get Blueprint Template Details</td>
<td>239</td>
</tr>
<tr>
<td>20.3</td>
<td>Add Blueprint Template</td>
<td>240</td>
</tr>
<tr>
<td>20.4</td>
<td>Edit Blueprint Template</td>
<td>241</td>
</tr>
<tr>
<td>20.5</td>
<td>Delete Blueprint Template</td>
<td>242</td>
</tr>
<tr>
<td>21</td>
<td>Catalogs</td>
<td></td>
</tr>
<tr>
<td>21.1</td>
<td>Get List of Catalogs</td>
<td>244</td>
</tr>
<tr>
<td>21.2</td>
<td>Get Catalog Details</td>
<td>245</td>
</tr>
<tr>
<td>21.3</td>
<td>Create Catalog</td>
<td>246</td>
</tr>
<tr>
<td>21.4</td>
<td>Add vApp to Catalog</td>
<td>246</td>
</tr>
<tr>
<td>21.5</td>
<td>Delete Catalog</td>
<td>247</td>
</tr>
<tr>
<td>21.6</td>
<td>Get List of vApp Templates</td>
<td>247</td>
</tr>
<tr>
<td>21.7</td>
<td>Get List of Media Files</td>
<td>248</td>
</tr>
</tbody>
</table>
21.8 Add File to Media Library ................................................................. 249
21.8.1 File uploading to the Control Panel ..................................................... 249
21.8.2 Adding file to media library .............................................................. 250
21.9 Create vApp Template (Add to Catalog) ................................................. 250
21.10 Delete vApp Template ........................................................................... 251

22 CDN Accelerator ..................................................................................... 252
22.1 Get List of Accelerators ......................................................................... 252
22.2 Get Accelerator Details ........................................................................... 256
22.3 Add Accelerator ...................................................................................... 260
22.4 Edit Accelerator ...................................................................................... 261
22.5 Reboot Accelerator .................................................................................. 262
22.6 Start up Accelerator ................................................................................ 262
22.7 Shut down Accelerator ............................................................................ 263
22.8 Suspend Accelerator ............................................................................... 263
22.9 Rebuild Accelerator ............................................................................... 263
22.10 Migrate Accelerator ............................................................................... 264
22.11 Delete Accelerator ............................................................................... 264
22.12 Unlock Accelerator ............................................................................... 265
22.13 Segregate Accelerator .......................................................................... 265
22.14 Change Accelerator Owner .................................................................... 266
22.15 Accelerator Network Interfaces ............................................................... 266
22.16 Accelerator IP Address Joins ................................................................. 268
22.17 View Accelerator Disks ......................................................................... 270
22.18 Rebuild Network for Accelerator ............................................................ 272
22.19 Get Accelerator CPU Usage Statistics .................................................. 272
## 23 CDN Edge Groups .......................................................... 274

23.1 Get List of CDN Edge Groups ........................................... 274
23.2 Get List of Available CDN Edge Groups .......................... 275
23.3 Get CDN Edge Group Details ........................................... 276
23.4 Add CDN Edge Group ..................................................... 277
23.5 Edit CDN Edge Group ..................................................... 278
23.6 Delete CDN Edge Group .................................................. 278
23.7 Assign Location to CDN Edge Group ............................... 279
23.8 Unassign Location From CDN Edge Group ..................... 279
23.9 Modify CDN Edge Group ................................................ 280
23.10 Search CDN Edge Groups ............................................. 280

## 24 CDN Edge Servers ......................................................... 282

24.1 Get List of CDN Edge Servers ........................................... 282
24.2 Get CDN Edge Server Details .......................................... 286
24.3 Add CDN Edge Server .................................................... 291
24.3.1 Page History ........................................................... 292
24.4 Edit CDN Edge Server .................................................... 292
24.5 Reboot CDN Edge Server ............................................... 293
24.6 Reboot CDN Edge Server in Recovery .............................. 293
24.7 Start up CDN Edge Server .............................................. 294
24.8 Shut down CDN Edge Server ......................................... 294
24.9 Stop CDN Edge Server .................................................. 294
24.10 Rebuild CDN Edge Server ............................................. 295
24.11 Suspend CDN Edge Server ........................................... 295
24.12 Rerun CDN Edge Server Creation Scripts ....................... 296
24.13 Unlock CDN Edge Server ................................................................. 296
24.14 Delete CDN Edge Server ................................................................. 296
24.15 Migrate CDN Edge Server ................................................................. 297
24.16 Segregate CDN Edge Server ............................................................. 297
24.17 Open CDN Edge Server Console ....................................................... 298
24.18 Change CDN Edge Server Owner ...................................................... 298
24.19 Set VIP Status for CDN Edge Server ................................................ 299
24.20 Add/Edit Admin/User Note for CDN Edge Server ............................... 299
24.21 CDN Edge Server Disks ................................................................... 300
24.22 CDN Edge Server Network Interfaces .............................................. 300
24.23 CDN Edge Server IP Address Joins .................................................... 302
24.24 Rebuild Network for CDN Edge Server ............................................. 304
24.25 Get CDN Edge Server Billing Statistics .......................................... 305
24.26 Get CDN Edge Server CPU Usage Statistics .................................... 309
24.27 Search CDN Edge Server by Label ................................................... 309

25 CDN HTTP Caching Rules ................................................................. 311
25.1 Get List of HTTP Caching Rules ....................................................... 311
25.2 Add HTTP Caching Rule ................................................................... 312
25.3 Edit HTTP Caching Rule ................................................................... 313
25.4 Delete HTTP Caching Rule ............................................................... 315

26 CDN Reporting .................................................................................. 316
26.1 CDN Top Files Report ..................................................................... 316
26.2 Purge CDN Resource in Top Files Report ......................................... 317
26.3 CDN Top Referrers Report ............................................................... 318
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>26.4</td>
<td>CDN Overview Report</td>
<td>319</td>
</tr>
<tr>
<td>26.5</td>
<td>CDN Cache Statistics Report</td>
<td>321</td>
</tr>
<tr>
<td>26.6</td>
<td>CDN Status Codes Report</td>
<td>322</td>
</tr>
<tr>
<td>26.7</td>
<td>CDN Bandwidth Statistics Report</td>
<td>324</td>
</tr>
<tr>
<td>27</td>
<td>CDN Resources</td>
<td>327</td>
</tr>
<tr>
<td>27.1</td>
<td>Get List of CDN Resources</td>
<td>327</td>
</tr>
<tr>
<td>27.2</td>
<td>Get CDN Resource Basic Details</td>
<td>329</td>
</tr>
<tr>
<td>27.3</td>
<td>Get CDN Resource Advanced Details</td>
<td>330</td>
</tr>
<tr>
<td>27.4</td>
<td>Add HTTP CDN Resource</td>
<td>331</td>
</tr>
<tr>
<td>27.4.1</td>
<td>Page history</td>
<td>333</td>
</tr>
<tr>
<td>27.5</td>
<td>Add Video on Demand CDN Resource</td>
<td>333</td>
</tr>
<tr>
<td>27.6</td>
<td>Add Live Streaming CDN Resource</td>
<td>334</td>
</tr>
<tr>
<td>27.7</td>
<td>Add HTTP Pull CDN Resource with Advanced Settings</td>
<td>336</td>
</tr>
<tr>
<td>27.7.1</td>
<td>Page history</td>
<td>341</td>
</tr>
<tr>
<td>27.8</td>
<td>Add HTTP Push CDN Resource with Advanced Settings</td>
<td>341</td>
</tr>
<tr>
<td>27.8.1</td>
<td>Page History</td>
<td>344</td>
</tr>
<tr>
<td>27.9</td>
<td>Add VoD Push CDN Resource With Advanced Settings</td>
<td>345</td>
</tr>
<tr>
<td>27.10</td>
<td>Add VoD Pull CDN Resource With Advanced Settings</td>
<td>347</td>
</tr>
<tr>
<td>27.11</td>
<td>Add Live Streaming CDN Resource with Advanced Settings</td>
<td>349</td>
</tr>
<tr>
<td>27.12</td>
<td>Edit CDN Resource</td>
<td>352</td>
</tr>
<tr>
<td>27.13</td>
<td>Edit HTTP Pull CDN Resource with Advanced Settings</td>
<td>353</td>
</tr>
<tr>
<td>27.13.1</td>
<td>Page history</td>
<td>357</td>
</tr>
<tr>
<td>27.14</td>
<td>Edit HTTP Push CDN Resource with Advanced Settings</td>
<td>357</td>
</tr>
<tr>
<td>27.14.1</td>
<td>Page History</td>
<td>361</td>
</tr>
<tr>
<td>27.15</td>
<td>Edit VoD Push CDN Resource with Advanced Settings</td>
<td>361</td>
</tr>
<tr>
<td>27.16</td>
<td>Edit VoD Pull CDN Resource with Advanced Settings</td>
<td>363</td>
</tr>
<tr>
<td>27.17</td>
<td>Edit Live Streaming CDN Resource with Advanced Settings</td>
<td>365</td>
</tr>
</tbody>
</table>
27.18 Delete CDN Resource........................................................................................................... 367
27.19 Change CDN Resource FTP Password.................................................................................. 368
27.20 Prefetch CDN Resource Content ........................................................................................ 368
27.21 Purge CDN Resource Content .............................................................................................. 369
27.22 View CDN Resource Bandwidth Statistics ............................................................................ 371
27.23 View CDN Resource Streaming Statistics ............................................................................. 374
27.24 View CDN Resource Billing Statistics .................................................................................. 374
27.25 View CDN Resource Raw Log Configuration ......................................................................... 375
27.26 Edit CDN Resource Raw Log Configuration .......................................................................... 377
27.27 Get Instruction for Live Streaming CDN Internal Resource ................................................. 378
27.28 Search CDN Resource ........................................................................................................ 380
27.29 Suspend CDN Resource ...................................................................................................... 381
27.30 Resume CDN Resource ........................................................................................................ 382
27.31 View CDN Advanced Reporting ........................................................................................... 382
27.32 Get List of Available Storage Locations ................................................................................ 384

28 CDN SSL Certificates ................................................................. 386
28.1 Get List of Custom SNI SSL Certificates .............................................................................. 387
28.2 Get Custom SNI SSL Certificate Details ............................................................................... 388
28.3 Add Custom SNI SSL Certificate .......................................................................................... 389
28.4 Edit Custom SNI SSL Certificate .......................................................................................... 389
28.5 Delete Custom SNI SSL Certificate ....................................................................................... 389
28.6 Add Custom SNI SSL Certificate to CDN Resource ............................................................. 389

29 CDN Storage Servers ................................................................. Error! Bookmark not defined.
29.1 Get List of CDN Storage Servers .......................................................................................... Error! Bookmark not defined.
29.2 Get List of HTTP Storage Servers ....................................................... Error! Bookmark not defined.
29.3 Get List of Streaming Storage Servers .............................................. Error! Bookmark not defined.
29.4 Get CDN Storage Server Details ......................................................... Error! Bookmark not defined.
29.5 Add CDN Storage Server ................................................................. Error! Bookmark not defined.
29.5.1 Page History .................................................................................... Error! Bookmark not defined.
29.6 Edit CDN Storage Server ................................................................. Error! Bookmark not defined.
29.7 Reboot CDN Storage Server ............................................................. Error! Bookmark not defined.
29.8 Start up CDN Storage Server ............................................................ Error! Bookmark not defined.
29.9 Shut down CDN Storage Server ....................................................... Error! Bookmark not defined.
29.10 Stop CDN Storage Server ................................................................ Error! Bookmark not defined.
29.11 Rebuild CDN Storage Server .......................................................... Error! Bookmark not defined.
29.12 Suspend CDN Storage Server .......................................................... Error! Bookmark not defined.
29.13 Unlock CDN Storage Server ............................................................. Error! Bookmark not defined.
29.14 Set VIP Status for CDN Storage Server ......................................... Error! Bookmark not defined.
29.15 Delete CDN Storage Server ............................................................. Error! Bookmark not defined.
29.16 Migrate CDN Storage Server .......................................................... Error! Bookmark not defined.
29.17 Segregate CDN Storage Server ....................................................... Error! Bookmark not defined.
29.18 Resize CDN Storage Server ............................................................. Error! Bookmark not defined.
29.19 Change CDN Storage Server Owner ............................................... Error! Bookmark not defined.
29.20 Rebuild Network for CDN Storage Server ....................................... Error! Bookmark not defined.
29.21 CDN Storage Server Disks ............................................................... Error! Bookmark not defined.
29.22 CDN Storage Server Network Interfaces ........................................ Error! Bookmark not defined.
29.23 CDN Storage Server IP Address Joins ............................................ Error! Bookmark not defined.
29.25 Get CDN Storage Server Hourly Statistics ...................................... Error! Bookmark not defined.
29.27 Search CDN Storage Server by Label ............................................ Error! Bookmark not defined.
29.28 Get List of CDN Storage Server Backups .......................... Error! Bookmark not defined.
29.29 Add/Edit Admin/User Note for CDN Storage Server .......... Error! Bookmark not defined.

30 CDN Usage Statistics ....Error! Bookmark not defined.

31 Check Password Strength .Error! Bookmark not defined.

32 CloudBoot IP Addresses ..Error! Bookmark not defined.
32.1 Get List of CloudBoot IP Addresses ........................................ Error! Bookmark not defined.
32.2 Add CloudBoot IP Address .............................................. Error! Bookmark not defined.
32.3 Edit CloudBoot IP Address .............................................. Error! Bookmark not defined.
32.4 Delete Cloud Boot IP Address ........................................ Error! Bookmark not defined.

33 Company Billing Plans ...Error! Bookmark not defined.
33.1 Get List of Company Billing Plans .......................... Error! Bookmark not defined.
33.2 Get Company Billing Plan Details .................................... Error! Bookmark not defined.
33.3 Get Company Billing Plan Resources ................................ Error! Bookmark not defined.
33.3.1 Page history ........................................................................ Error! Bookmark not defined.
33.4 Get Company Billing Plan Resource Details .................... Error! Bookmark not defined.
33.4.1 Page history ........................................................................ Error! Bookmark not defined.
33.5 Create Company Billing Plan ............................................ Error! Bookmark not defined.
33.6 Add Limits for Compute Zone ........................................ Error! Bookmark not defined.
33.6.1 Page history ........................................................................ Error! Bookmark not defined.
33.7 Add Limits for Data Store Zones .................................... Error! Bookmark not defined.
33.7.1 Page history ........................................................................ Error! Bookmark not defined.
33.8 Add Limits for Network Zone ........................................ Error! Bookmark not defined.
33.9 Edit Company Billing Plan .............................................. Error! Bookmark not defined.
33.10 Edit Limits & Pricing for Data Store Zones in Company Billing Plan Error! Bookmark not defined.
  33.10.1 Page history Error! Bookmark not defined.

33.11 Edit Limits & Pricing for Network Zones in Company Billing Plan Error! Bookmark not defined.

  33.12.1 Page history Error! Bookmark not defined.

33.13 Delete Resource From Company Billing Plan Error! Bookmark not defined.

33.14 Delete Company Billing Plan Error! Bookmark not defined.

34 Company Payments Error! Bookmark not defined.
  34.1 Get List of Company Payments Error! Bookmark not defined.
  34.2 Get List of Company Monthly Bills Error! Bookmark not defined.
  34.3 Create Company Payment Error! Bookmark not defined.
  34.4 Edit Company Payment Error! Bookmark not defined.
  34.5 Delete Company Payment Error! Bookmark not defined.

35 Compute Resources Error! Bookmark not defined.
  35.1 Get List of Compute Resources Error! Bookmark not defined.
  35.1.1 Page History Error! Bookmark not defined.
  35.2 Get Compute Resource Details Error! Bookmark not defined.
  35.2.1 Page History Error! Bookmark not defined.
  35.3 Add Xen/KVM Compute Resource Error! Bookmark not defined.
  35.4 Add CloudBoot Compute Resource Error! Bookmark not defined.
  35.4.1 Page history Error! Bookmark not defined.
  35.5 Add Smart CloudBoot Compute Resource Error! Bookmark not defined.
  35.5.1 Page history Error! Bookmark not defined.
  35.6 Add Baremetal CloudBoot Compute Resource Error! Bookmark not defined.
  35.6.1 Page history Error! Bookmark not defined.
  35.7 Add VMware Compute Resource Error! Bookmark not defined.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>35.8</td>
<td>Edit Xen/KVM Compute Resource</td>
</tr>
<tr>
<td>35.9</td>
<td>Edit CloudBoot Compute Resource</td>
</tr>
<tr>
<td>35.10</td>
<td>Edit Smart CloudBoot Compute Resource</td>
</tr>
<tr>
<td>35.11</td>
<td>Edit Baremetal CloudBoot Compute Resource</td>
</tr>
<tr>
<td>35.12</td>
<td>Reboot Compute Resource</td>
</tr>
<tr>
<td>35.13</td>
<td>Delete Compute Resource</td>
</tr>
<tr>
<td>35.14</td>
<td>Edit VMware Compute Resource</td>
</tr>
<tr>
<td>35.15</td>
<td>Get List of Appliances Running on Compute Resource</td>
</tr>
<tr>
<td>35.16</td>
<td>Get List of Data Store Joins Attached to Compute Resource</td>
</tr>
<tr>
<td>35.17</td>
<td>Get List of Data Stores Attached to Compute Resource</td>
</tr>
<tr>
<td>35.18</td>
<td>Add Data Store Join to Compute Resource</td>
</tr>
<tr>
<td>35.19</td>
<td>Remove Data Store Join from Compute Resource</td>
</tr>
<tr>
<td>35.20</td>
<td>Get List of Compute Resource Network Joins</td>
</tr>
<tr>
<td>35.21</td>
<td>Add Network Join to Compute Resource</td>
</tr>
<tr>
<td>35.22</td>
<td>Remove Network Join from Compute Resource</td>
</tr>
<tr>
<td>35.23</td>
<td>Enable/disable Open vSwitch</td>
</tr>
<tr>
<td>35.24</td>
<td>Power Cycle CloudBoot Compute Resource</td>
</tr>
<tr>
<td>35.25</td>
<td>Enable Maintenance Mode for Xen/KVM Compute Resource</td>
</tr>
<tr>
<td>35.26</td>
<td>Disable Maintenance Mode for Xen/KVM Compute Resource</td>
</tr>
<tr>
<td>35.27</td>
<td>Add Backup Server to Compute Resource</td>
</tr>
<tr>
<td>35.28</td>
<td>Remove Backup Server from Compute Resource</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>36</td>
<td>Compute Zones</td>
</tr>
<tr>
<td>36.1</td>
<td>Get List of Compute Zones</td>
</tr>
<tr>
<td>36.1.1</td>
<td>Page history</td>
</tr>
<tr>
<td>36.2</td>
<td>Get Compute Zone Details</td>
</tr>
<tr>
<td>36.2.1</td>
<td>Page history</td>
</tr>
</tbody>
</table>
36.3 Add Compute Zone ................................................................. Error! Bookmark not defined.
36.3.1 Page history ................................................................. Error! Bookmark not defined.

36.4 Edit Compute Zone ............................................................. Error! Bookmark not defined.
36.4.1 Page history ................................................................. Error! Bookmark not defined.

36.5 Delete Compute Zone .......................................................... Error! Bookmark not defined.

36.6 Get List of Compute Resources Attached to Compute Zone Error! Bookmark not defined.

36.7 Attach Compute Resource to Compute Zone ............................. Error! Bookmark not defined.

36.8 Remove Compute Resource from Compute Zone ....................... Error! Bookmark not defined.

36.9 Get the List of Data Store Joins Attached to Compute Zone Error! Bookmark not defined.

36.10 Add Data Store Join to Compute Zone ................................. Error! Bookmark not defined.

36.11 Remove Data Store Join from Compute Zone .......................... Error! Bookmark not defined.

36.12 Get List of Network Joins Attached to Compute Zone .............. Error! Bookmark not defined.

36.13 Add Network Join to Compute Zone ...................................... Error! Bookmark not defined.

36.14 Remove Network Join from Compute Zone ............................. Error! Bookmark not defined.

36.15 Update CPU Flags for Compute Zone .................................... Error! Bookmark not defined.

36.16 Get Common CPU Flags for Compute Resources of Compute Zone Error! Bookmark not defined.

36.17 Add Backup Server to Compute Zone .................................... Error! Bookmark not defined.

36.18 Remove Backup Server from Compute Zone ............................. Error! Bookmark not defined.

37 Container Servers ........ Error! Bookmark not defined.

37.1 Get List of All Container Servers ......................................... Error! Bookmark not defined.

37.2 Get Container Server Details .............................................. Error! Bookmark not defined.

37.3 Get Statuses for all Container Servers .................................. Error! Bookmark not defined.

37.4 Get Container Server Status .............................................. Error! Bookmark not defined.

37.5 Get Container Server Cloud Config ...................................... Error! Bookmark not defined.

37.6 Add Container Server ......................................................... Error! Bookmark not defined.
37.7 Add Container Server Cloud Config
37.8 View Encrypted Container Server Password
37.9 Build or Rebuild Container Server
37.10 Edit Container Server
37.11 Edit Container Server Cloud Config
37.12 Change Container Server Owner
37.13 Reset Container Server Root Password
37.14 Migrate Container Server
37.15 Set VIP Status for Container Server
37.16 Delete Container Server
37.17 Start up Container Server
37.18 Segregate Container Server
37.19 Desegregate Container Server
37.20 Reboot Container Server
37.21 Reboot Container Server in Recovery
37.22 Boot Container Server from ISO
37.23 Suspend Container Server
37.24 Unlock Container Server
37.25 Unsuspend Container Server
37.26 Shut down Container Server
37.27 Stop Container Server
37.28 Open Container Server Console
37.29 Container Server Billing Statistics
37.30 Search Container Server by Label
37.31 Get Container Server CPU Usage Statistics
37.32  Add/Edit Admin/User Note for Container Server....................... Error! Bookmark not defined.
37.32.1 Add/Edit User Note................................................................. Error! Bookmark not defined.

38  Control Panel Maintenance ........ Error! Bookmark not defined.
38.1 Get Control Panel Maintenance Status ....................................... Error! Bookmark not defined.
38.2 Enable Control Panel Maintenance ............................................ Error! Bookmark not defined.
38.3 Disable Control Panel Maintenance .......................................... Error! Bookmark not defined.

39  Currencies ............... Error! Bookmark not defined.
39.1 Get List of Currencies ............................................................... Error! Bookmark not defined.
39.2 Get Currency Details ............................................................... Error! Bookmark not defined.
39.3 Add Currency ............................................................................ Error! Bookmark not defined.
39.4 Edit Currency ............................................................................ Error! Bookmark not defined.
39.5 Delete Currency ........................................................................ Error! Bookmark not defined.

40  Custom Recipe Variables . Error! Bookmark not defined.
40.1 Get List of Custom Variables ...................................................... Error! Bookmark not defined.
40.2 Get Custom Variable Details ....................................................... Error! Bookmark not defined.
40.3 Edit Custom Variable ................................................................. Error! Bookmark not defined.
40.4 Add Custom Variable ................................................................. Error! Bookmark not defined.
40.5 Delete Custom Variable .............................................................. Error! Bookmark not defined.
40.6 Get List of Virtual Server Custom Variables ............................... Error! Bookmark not defined.
40.7 Get List of Smart Server Custom Variables ................................. Error! Bookmark not defined.
40.8 Get List of Baremetal Server Variables ...................................... Error! Bookmark not defined.
40.9 Get Virtual Server Custom Variable Details ............................... Error! Bookmark not defined.
40.10 Get Smart Server Custom Variable Details ............................... Error! Bookmark not defined.
40.11 Get Baremetal Server Custom Variable Details.............. Error! Bookmark not defined.

40.12 Add Virtual Server Custom Variable.......................... Error! Bookmark not defined.

40.13 Add Smart Server Custom Variable............................. Error! Bookmark not defined.

40.14 Add Baremetal Server Custom Variable......................... Error! Bookmark not defined.

40.15 Edit Virtual Server Custom Variable............................. Error! Bookmark not defined.

40.16 Edit Smart Server Custom Variable............................. Error! Bookmark not defined.

40.17 Edit Baremetal Server Custom Variable......................... Error! Bookmark not defined.

40.18 Delete Virtual Server Custom Variable......................... Error! Bookmark not defined.

40.19 Delete Smart Server Custom Variable......................... Error! Bookmark not defined.

40.20 Delete Baremetal Server Custom Variable...................... Error! Bookmark not defined.

41 Data Stores .............. Error! Bookmark not defined.

41.1 Get List of Data Stores............................................ Error! Bookmark not defined.

41.2 Get Data Store Details.............................................. Error! Bookmark not defined.

41.3 Add LVM Data Store................................................ Error! Bookmark not defined.

41.4 Add VMware Data Store............................................. Error! Bookmark not defined.

41.5 Add SolidFire Data Store.......................................... Error! Bookmark not defined.

41.6 Edit LVM Data Store................................................ Error! Bookmark not defined.
41.6.1 Page history ....................................................... Error! Bookmark not defined.

41.7 Edit SolidFire Data Store.......................................... Error! Bookmark not defined.

41.8 Delete Data Store.................................................... Error! Bookmark not defined.

42 Data Store Zones ........ Error! Bookmark not defined.

42.1 Get List of Data Store Zones....................................... Error! Bookmark not defined.
42.1.1 Page history .......................................................... Error! Bookmark not defined.

42.2 Get Data Store Zone Details........................................ Error! Bookmark not defined.
42.2.1 Page history .......................................................... Error! Bookmark not defined.
42.3 Add Data Store Zone .......................................................... Error! Bookmark not defined.
42.3.1 Page history ............................................................... Error! Bookmark not defined.
42.4 Edit Data Store Zone .......................................................... Error! Bookmark not defined.
42.4.1 Page history ............................................................... Error! Bookmark not defined.
42.5 Delete Data Store Zone .......................................................... Error! Bookmark not defined.
42.6 Get the List of Data Stores Attached to Data Store Zone ...... Error! Bookmark not defined.
42.7 Attach Data Store to Data Store Zone .................................. Error! Bookmark not defined.
42.8 Detach Data Store from Data Store Zone ............................. Error! Bookmark not defined.

43 Disks .................................. Error! Bookmark not defined.
43.1 Get List of Disks ............................................................. Error! Bookmark not defined.
43.2 Get List of VS Disks .......................................................... Error! Bookmark not defined.
43.3 Get VS Disk Details ......................................................... Error! Bookmark not defined.
43.4 Add New Disk ............................................................... Error! Bookmark not defined.
43.5 Edit Disk ................................................................. Error! Bookmark not defined.
43.6 Migrate Disk ............................................................... Error! Bookmark not defined.
43.7 Delete Disk ............................................................... Error! Bookmark not defined.
43.8 View Disk IOPS ............................................................. Error! Bookmark not defined.
43.9 Build Disk ............................................................... Error! Bookmark not defined.
43.10 Unlock Disk ............................................................... Error! Bookmark not defined.
43.11 Get List of Backups Available for a Disk ............................. Error! Bookmark not defined.

44 DNS Setup ....................... Error! Bookmark not defined.
44.1 Get DNS Domain Details .................................................. Error! Bookmark not defined.
44.2 Set Up DNS Domain ....................................................... Error! Bookmark not defined.
44.3 Edit DNS Domain .......................................................... Error! Bookmark not defined.
44.4 Get List of Glue Records .................................................. Error! Bookmark not defined.
DNS Zones

45.1 Get List of Own DNS Zones
45.2 Get Domain Zone Details
45.3 Get List of Users DNS Zones
45.4 Add DNS Zone
45.5 Delete DNS Zone
45.6 Get List of Name Servers
45.7 Get List of DNS Zone Records
45.8 Get DNS Record Details
45.9 Add DNS Record
45.10 Edit DNS Records
45.11 Delete DNS Record

Edge Gateways

46.1 Get List of Edge Gateways
46.2 Get List of Edge Gateway Interfaces
46.3 Get Edge Gateway Details
46.4 Create Edge Gateway
46.5 Edit Edge Gateway
46.6 Delete Edge Gateway

Embed Statistics Charts

Errors

48.1 Get List of Errors
48.2  Get Error Details ............................................................. Error! Bookmark not defined.

49  Federation  ................................................................. Error! Bookmark not defined.
49.1  Add Zone to Federation .................................................... Error! Bookmark not defined.
49.2  Enable Federated Zone ...................................................... Error! Bookmark not defined.
49.3  Disable Federated Zone .................................................... Error! Bookmark not defined.
49.4  Remove Zone from Federation .......................................... Error! Bookmark not defined.
49.5  Get List of Federated Resources ...................................... Error! Bookmark not defined.
49.6  Get Federated Resource Details ...................................... Error! Bookmark not defined.
49.7  Subscribe to Federated Zone ............................................ Error! Bookmark not defined.
49.8  Unsubscribe from Federated Zone ...................................... Error! Bookmark not defined.
49.9  Suspend Zone ............................................................ Error! Bookmark not defined.
49.10  Unsuspend Zone .......................................................... Error! Bookmark not defined.

50  Firewall Rules for VSs ..................................................... Error! Bookmark not defined.
50.1  Get List of Firewall Rules ............................................... Error! Bookmark not defined.
50.2  Apply Firewall Rule ....................................................... Error! Bookmark not defined.
50.3  Add Firewall Rule ........................................................ Error! Bookmark not defined.
50.4  Edit Firewall Rule ........................................................ Error! Bookmark not defined.
50.5  Delete Firewall Rule ...................................................... Error! Bookmark not defined.
50.6  Change Firewall Rule Position ........................................ Error! Bookmark not defined.
50.7  Set Default Firewall Rules ............................................. Error! Bookmark not defined.

51  Firewalls  ................................................................. Error! Bookmark not defined.
51.1  Get List of Firewalls ..................................................... Error! Bookmark not defined.
51.2  Get Firewall Details ...................................................... Error! Bookmark not defined.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>51.3</td>
<td>Add Firewall</td>
</tr>
<tr>
<td>51.4</td>
<td>Edit Firewall</td>
</tr>
<tr>
<td>51.5</td>
<td>Delete Firewall</td>
</tr>
<tr>
<td>52</td>
<td><strong>High Availability Control Panel</strong></td>
</tr>
<tr>
<td>52.1</td>
<td>Get List of Clusters</td>
</tr>
<tr>
<td>52.2</td>
<td>Get List of Cluster Nodes</td>
</tr>
<tr>
<td>52.3</td>
<td>Get Node Details</td>
</tr>
<tr>
<td>52.4</td>
<td>Get List of Hosts</td>
</tr>
<tr>
<td>52.5</td>
<td>Get Host Nodes</td>
</tr>
<tr>
<td>52.6</td>
<td>Get List of Communication Rings</td>
</tr>
<tr>
<td>52.7</td>
<td>Get Details of Communication Ring</td>
</tr>
<tr>
<td>52.8</td>
<td>Get Status of OnApp Subsystems</td>
</tr>
<tr>
<td>52.9</td>
<td>Enable High Availability</td>
</tr>
<tr>
<td>52.10</td>
<td>Disable High Availability</td>
</tr>
<tr>
<td>52.11</td>
<td>Deactivate Cluster</td>
</tr>
<tr>
<td>52.12</td>
<td>Activate Deactivated Cluster</td>
</tr>
<tr>
<td>52.13</td>
<td>Apply Changes to High Availability Configuration</td>
</tr>
<tr>
<td>52.14</td>
<td>Apply Changes to Multicast Configuration</td>
</tr>
<tr>
<td>52.15</td>
<td>Edit Host</td>
</tr>
<tr>
<td>52.16</td>
<td>Edit Cluster</td>
</tr>
<tr>
<td>52.17</td>
<td>Edit Node</td>
</tr>
<tr>
<td>52.18</td>
<td>Edit Communication Ring</td>
</tr>
<tr>
<td>52.19</td>
<td>Add Cluster</td>
</tr>
<tr>
<td>52.20</td>
<td>Add Host</td>
</tr>
</tbody>
</table>
### 52.21 Add New Node to Cluster

Error! Bookmark not defined.

### 52.22 Add Communication Interface

Error! Bookmark not defined.

### 52.23 Delete Host

Error! Bookmark not defined.

### 52.24 Delete Node

Error! Bookmark not defined.

### 52.25 Delete Communication Ring

Error! Bookmark not defined.

---

### 53 Instance Packages

Error! Bookmark not defined.

#### 53.1 Get List of Instance Packages

Error! Bookmark not defined.

#### 53.2 Get Instance Package Details

Error! Bookmark not defined.

#### 53.3 Add Instance Package

Error! Bookmark not defined.

#### 53.4 Edit Instance Package

Error! Bookmark not defined.

#### 53.5 Delete Instance Package

Error! Bookmark not defined.

---

### 54 Integrated Storage

Error! Bookmark not defined.

#### 54.1 Get List of Integrated Storage Data Stores

Error! Bookmark not defined.

#### 54.2 Get Integrated Data Store Details

Error! Bookmark not defined.

#### 54.3 Add Integrated Storage Data Store

Error! Bookmark not defined.

##### 54.3.1 PAGE HISTORY

Error! Bookmark not defined.

#### 54.4 Edit Integrated Data Store

Error! Bookmark not defined.

##### 54.4.1 PAGE HISTORY

Error! Bookmark not defined.

#### 54.5 Delete Integrated Storage Data Store

Error! Bookmark not defined.

##### 54.5.1 PAGE HISTORY

Error! Bookmark not defined.

#### 54.6 Add Disk Drive to Integrated Storage Data Store

Error! Bookmark not defined.

##### 54.6.1 PAGE HISTORY

Error! Bookmark not defined.

#### 54.7 Remove Disk Drive from Integrated Storage Data Store

Error! Bookmark not defined.

##### 54.7.1 PAGE HISTORY

Error! Bookmark not defined.

#### 54.8 Get Storage Node IO Statistics

Error! Bookmark not defined.

#### 54.9 Get Integrated Storage Datastore Disk IO Statistics

Error! Bookmark not defined.
54.10 Forget Storage Node ................................................................. Error! Bookmark not defined.
54.11 Get List of Backend Nodes on Integrated Storage Data Store .......... Error! Bookmark not defined.
54.12 Get List of Integrated Storage Data Stores Disk Drives .......... Error! Bookmark not defined.

55 IP Addresses ............ Error! Bookmark not defined.
55.1 Get List of Network IP Addresses ................................................. Error! Bookmark not defined.
55.2 Add IP Address Record ............................................................. Error! Bookmark not defined.
55.3 Edit IP Address ........................................................................ Error! Bookmark not defined.
55.4 Delete IP address ......................................................................... Error! Bookmark not defined.
55.5 Assign IP Address to User .......................................................... Error! Bookmark not defined.
55.6 Unassign IP Address from User .................................................. Error! Bookmark not defined.

56 IP Address Joins ......... Error! Bookmark not defined.
56.1 Get List of IP Address Joins ....................................................... Error! Bookmark not defined.
56.2 Assign IP Address Join to VS ..................................................... Error! Bookmark not defined.
56.3 Delete IP Address Join ............................................................... Error! Bookmark not defined.

57 ISOs ....................... Error! Bookmark not defined.
57.1 Get List of ISOs ......................................................................... Error! Bookmark not defined.
57.2 Get ISO Details ........................................................................ Error! Bookmark not defined.
57.3 Get List of Public ISOs .............................................................. Error! Bookmark not defined.
57.4 Get List of ISOs of Particular User ............................................. Error! Bookmark not defined.
57.5 Get List of User ISOs ................................................................. Error! Bookmark not defined.
57.6 Get List of Own ISOs ................................................................. Error! Bookmark not defined.
57.7 Update ISO .............................................................................. Error! Bookmark not defined.
57.8 Add New ISO ............................................................................ Error! Bookmark not defined.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>57.9</td>
<td>Make ISO Public</td>
</tr>
<tr>
<td>57.10</td>
<td>Delete ISO</td>
</tr>
<tr>
<td>58</td>
<td>OVAs</td>
</tr>
<tr>
<td>58.1</td>
<td>Get List of OVAs</td>
</tr>
<tr>
<td>58.2</td>
<td>Get List of System OVAs</td>
</tr>
<tr>
<td>58.3</td>
<td>Get List of Own OVAs</td>
</tr>
<tr>
<td>58.4</td>
<td>Get List of User OVAs</td>
</tr>
<tr>
<td>58.5</td>
<td>Get OVA Details</td>
</tr>
<tr>
<td>58.6</td>
<td>Upload OVA</td>
</tr>
<tr>
<td>58.7</td>
<td>Unlock OVA</td>
</tr>
<tr>
<td>58.8</td>
<td>Edit OVA</td>
</tr>
<tr>
<td>58.9</td>
<td>Make OVA Public</td>
</tr>
<tr>
<td>58.10</td>
<td>Search OVA</td>
</tr>
<tr>
<td>58.11</td>
<td>Delete OVA Files</td>
</tr>
<tr>
<td>58.12</td>
<td>Delete OVA</td>
</tr>
<tr>
<td>59</td>
<td>License</td>
</tr>
<tr>
<td>59.1</td>
<td>Get License Details</td>
</tr>
<tr>
<td>59.2</td>
<td>Edit License Details</td>
</tr>
<tr>
<td>60</td>
<td>Load Balancers</td>
</tr>
<tr>
<td>60.1</td>
<td>Get List of Load Balancers</td>
</tr>
<tr>
<td>60.2</td>
<td>Get Load Balancer Details</td>
</tr>
<tr>
<td>60.3</td>
<td>Get the List of Load Balancing Clusters</td>
</tr>
<tr>
<td>60.4</td>
<td>Get Load Balancing Cluster Details</td>
</tr>
</tbody>
</table>
60.5 Get Load Balancer Billing Statistics ........................................ Error! Bookmark not defined.
60.6 Get List of Load Balancer Autoscaling Monitors .................... Error! Bookmark not defined.
60.7 Get Load Balancer Autoscaling Monitor Details ..................... Error! Bookmark not defined.
60.8 Add Load Balancing Cluster ................................................. Error! Bookmark not defined.
60.9 Add Autoscaling Cluster ...................................................... Error! Bookmark not defined.
60.10 Add Nodes to Cluster Type .................................................. Error! Bookmark not defined.
60.11 Remove Nodes from Cluster Type ........................................ Error! Bookmark not defined.
60.12 Edit Load Balancing Cluster ............................................... Error! Bookmark not defined.
60.13 Edit Autoscaling Cluster ..................................................... Error! Bookmark not defined.
60.14 Edit Load Balancing Cluster Ports ..................................... Error! Bookmark not defined.
60.15 Delete Load Balancing Cluster ............................................. Error! Bookmark not defined.
60.16 Rebuild Load Balancer ....................................................... Error! Bookmark not defined.
60.17 Search Load Balancer by Label ......................................... Error! Bookmark not defined.
60.18 Start up Load Balancer ....................................................... Error! Bookmark not defined.
60.19 Shut down Load Balancer .................................................... Error! Bookmark not defined.
60.20 Suspend Load Balancer ....................................................... Error! Bookmark not defined.
60.21 Stop Load Balancer ............................................................ Error! Bookmark not defined.
60.22 Unlock Load Balancer ......................................................... Error! Bookmark not defined.

61 Locales .........................................................Error! Bookmark not defined.

62 Location Groups ...............Error! Bookmark not defined.
62.1 Get List of Location Groups ................................................ Error! Bookmark not defined.
62.2 Get Location Group Details ................................................ Error! Bookmark not defined.
62.3 Refresh Location Groups ..................................................... Error! Bookmark not defined.
62.4 Attach Compute Zone to Location Group ............................... Error! Bookmark not defined.
62.5 Detach Compute Zone from Location Group............................................ Error! Bookmark not defined.
62.6 Attach Data Store Zone to Location Group............................................. Error! Bookmark not defined.
62.7 Detach Data Store Zone from Location Group......................................... Error! Bookmark not defined.
62.8 Attach Network Zone to Location Group................................................ Error! Bookmark not defined.
62.9 Detach Network Zone from Location Group............................................. Error! Bookmark not defined.
62.10 Attach Backup Server Zone to Location Group....................................... Error! Bookmark not defined.
62.11 Detach Backup Server Zone from Location Group.................................. Error! Bookmark not defined.
62.12 Get List of Compute Zones Attached to Location Group............... Error! Bookmark not defined.
62.13 Get List of Data Store Zones Attached to Location Group.................... Error! Bookmark not defined.
62.14 Get List of Network Zones Attached to Location Group...................... Error! Bookmark not defined.
62.15 Get List of Backup Server Zones Attached to Location Group.............. Error! Bookmark not defined.
62.16 Attach CDN Location to Location Group............................................... Error! Bookmark not defined.
62.17 Detach CDN Location from Location Group........................................... Error! Bookmark not defined.
62.18 Get List of CDN Locations..................................................................... Error! Bookmark not defined.
62.19 Get List of CDN Locations for Location Group................................. Error! Bookmark not defined.

63 Logs .................................................. Error! Bookmark not defined.
63.1 Get List of Log Items.................................................................................. Error! Bookmark not defined.
63.2 Get Log Item Details................................................................................... Error! Bookmark not defined.
63.3 Get List of VS Log Items.......................................................................... Error! Bookmark not defined.
63.4 Get VS Log Item Details............................................................................ Error! Bookmark not defined.
63.5 Get Resource Difference Details.............................................................. Error! Bookmark not defined.

64 My Template Groups ......... Error! Bookmark not defined.
64.1 Get My Template Groups List.................................................................... Error! Bookmark not defined.
64.2 Get My Template Group Details............................................................... Error! Bookmark not defined.
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>64.3</td>
<td>Add My Template Group</td>
</tr>
<tr>
<td>64.4</td>
<td>Add Child Group</td>
</tr>
<tr>
<td>64.5</td>
<td>Edit My Template Group</td>
</tr>
<tr>
<td>64.6</td>
<td>Delete My Template Group</td>
</tr>
<tr>
<td>64.7</td>
<td>Get List of Templates Attached to Group</td>
</tr>
<tr>
<td>64.8</td>
<td>Attach Template to Group</td>
</tr>
<tr>
<td>64.9</td>
<td>Detach Template from Group</td>
</tr>
<tr>
<td>65</td>
<td>Network Interfaces</td>
</tr>
<tr>
<td>65.1</td>
<td>Get List of VS Network Interfaces</td>
</tr>
<tr>
<td>65.2</td>
<td>Get Network Interface Details</td>
</tr>
<tr>
<td>65.3</td>
<td>Add Network Interface to VS</td>
</tr>
<tr>
<td>65.4</td>
<td>Edit Network Interface</td>
</tr>
<tr>
<td>65.5</td>
<td>Delete Network Interface</td>
</tr>
<tr>
<td>65.6</td>
<td>Get VS Network Interface Usage Statistics</td>
</tr>
<tr>
<td>66</td>
<td>Networks</td>
</tr>
<tr>
<td>66.1</td>
<td>Get List of Networks</td>
</tr>
<tr>
<td>66.2</td>
<td>Get Network Details</td>
</tr>
<tr>
<td>66.3</td>
<td>Add Network</td>
</tr>
<tr>
<td>66.4</td>
<td>Edit Network</td>
</tr>
<tr>
<td>66.5</td>
<td>Rebuild VS Network</td>
</tr>
<tr>
<td>66.6</td>
<td>Delete Network</td>
</tr>
<tr>
<td>67</td>
<td>Network Zones</td>
</tr>
<tr>
<td>67.1</td>
<td>Get List of Network Zones</td>
</tr>
<tr>
<td>67.1.1</td>
<td>Page history</td>
</tr>
</tbody>
</table>

Error! Bookmark not defined.
67.2 Get Network Zone Details ............................................................... Error! Bookmark not defined.
67.2.1 Page history ................................................................................. Error! Bookmark not defined.

67.3 Add Network Zone ........................................................................ Error! Bookmark not defined.
67.3.1 Page history ................................................................................. Error! Bookmark not defined.

67.4 Edit Network Zone ......................................................................... Error! Bookmark not defined.
67.4.1 Page history ................................................................................. Error! Bookmark not defined.

67.5 Delete Network Zone ...................................................................... Error! Bookmark not defined.
67.6 Attach Network to Network Zone .................................................. Error! Bookmark not defined.
67.7 Remove Network from Network Zone ........................................... Error! Bookmark not defined.

68 OnApp Engine ................................................................. Error! Bookmark not defined.
68.1 Get OnApp Engine Status ............................................................. Error! Bookmark not defined.
68.2 Start OnApp Engine ....................................................................... Error! Bookmark not defined.
68.3 Stop OnApp Engine ....................................................................... Error! Bookmark not defined.
68.4 Reload OnApp Engine .................................................................... Error! Bookmark not defined.

69 Orchestration Models .... Error! Bookmark not defined.
69.1 Get List of Orchestration Models .................................................. Error! Bookmark not defined.
69.2 Get Orchestration Model Details ................................................... Error! Bookmark not defined.
69.3 Create Orchestration Model .......................................................... Error! Bookmark not defined.
69.4 Deploy Orchestration Model .......................................................... Error! Bookmark not defined.
69.5 Delete Orchestration Model .......................................................... Error! Bookmark not defined.

70 Pagination ......................... Error! Bookmark not defined.

71 Payments ......................... Error! Bookmark not defined.
71.1 Get List of All User Payments ...................................................... Error! Bookmark not defined.
71.1.1 Page history ................................................................................. Error! Bookmark not defined.
71.2 Get List of Payments of Particular User ....................................... Error! Bookmark not defined.
71.2.1 Page history ................................................................. Error! Bookmark not defined.

71.3 Get User Payment Details ............................................... Error! Bookmark not defined.
71.3.1 Page history ................................................................. Error! Bookmark not defined.

71.4 Create User Payment ...................................................... Error! Bookmark not defined.
71.5 Edit User Payment .......................................................... Error! Bookmark not defined.
71.6 Delete User Payment ....................................................... Error! Bookmark not defined.

72 Publishing Rules ........ Error! Bookmark not defined.
72.1 Get List of Publishing Rules .............................................. Error! Bookmark not defined.
72.2 Add Publishing Rule ........................................................ Error! Bookmark not defined.
72.3 Delete Publishing Rule ..................................................... Error! Bookmark not defined.

73 Recipe Groups ........ Error! Bookmark not defined.
73.1 Get List of Recipe Groups .................................................. Error! Bookmark not defined.
73.2 Get Recipe Group Details .................................................. Error! Bookmark not defined.
73.3 Add Recipe Group ............................................................ Error! Bookmark not defined.
73.4 Add Child Recipe Group .................................................... Error! Bookmark not defined.
73.5 Edit Recipe Group ............................................................ Error! Bookmark not defined.
73.6 Delete Recipe Group ........................................................ Error! Bookmark not defined.
73.7 Get List of Recipes Attached to Recipe Group ...................... Error! Bookmark not defined.
73.8 Attach Recipe to Recipe Group .......................................... Error! Bookmark not defined.
73.9 Remove Recipe from Recipe Group .................................... Error! Bookmark not defined.

74 Recipes ............... Error! Bookmark not defined.
74.1 Get List of Recipes .......................................................... Error! Bookmark not defined.
74.2 Get List of Servers Using Recipe ....................................... Error! Bookmark not defined.
74.3 Get Recipe Details ........................................................... Error! Bookmark not defined.
74.4 Add Recipe................................................................. Error! Bookmark not defined.
74.5 Edit Recipe........................................................................ Error! Bookmark not defined.
74.6 Delete Recipe...................................................................... Error! Bookmark not defined.
74.7 Manage Recipe Steps........................................................ Error! Bookmark not defined.
74.7.1 Get the List of Recipe Steps.......................................... Error! Bookmark not defined.
74.7.2 Create Recipe Step....................................................... Error! Bookmark not defined.
74.7.3 Edit Recipe Step.......................................................... Error! Bookmark not defined.
74.7.4 Remove Recipe Step.................................................... Error! Bookmark not defined.
74.7.5 Swap Recipe Steps Locations....................................... Error! Bookmark not defined.
74.8 Manage Virtual Server Recipes....................................... Error! Bookmark not defined.
74.8.1 Get the List of Virtual Server Recipes........................ Error! Bookmark not defined.
74.8.2 Assign Recipe to Virtual Server.................................. Error! Bookmark not defined.
74.8.3 Remove Recipe from Virtual Server.............................. Error! Bookmark not defined.
74.8.4 Run Recipe on Multiple Virtual Servers...................... Error! Bookmark not defined.
74.9 Manage Smart Server Recipes........................................ Error! Bookmark not defined.
74.9.1 Get the List of Smart Server Recipes............................ Error! Bookmark not defined.
74.9.2 Assign Recipe to Smart Server.................................... Error! Bookmark not defined.
74.9.3 Remove Recipe from Smart Server............................... Error! Bookmark not defined.
74.9.4 Run Recipe on Multiple Smart Servers....................... Error! Bookmark not defined.
74.10 Manage Baremetal Server Recipes................................ Error! Bookmark not defined.
74.10.1 Get the List of Baremetal Server Recipes................... Error! Bookmark not defined.
74.10.2 Assign Recipe to Baremetal Server............................. Error! Bookmark not defined.
74.10.3 Remove Recipe from Baremetal Server....................... Error! Bookmark not defined.
74.11 Manage Template Recipes............................................ Error! Bookmark not defined.
74.11.1 Get the List of Template Recipes................................ Error! Bookmark not defined.
74.11.2 Assign Recipe to Template......................................... Error! Bookmark not defined.
74.11.3 Remove Recipe from Template.................................. Error! Bookmark not defined.
74.12 Manage Compute Zone Recipes..................................... Error! Bookmark not defined.
74.12.1 Get the List of Compute Zone Recipes......................... Error! Bookmark not defined.
74.12.2 Assign Recipe to Compute Zone.................................. Error! Bookmark not defined.
74.12.3 Remove Recipe from Compute Zone......................... Error! Bookmark not defined.
74.13 Manage Control Panel Recipes...................................... Error! Bookmark not defined.
74.13.1 Get the List of Control Panel Recipes......................... Error! Bookmark not defined.
74.13.2 Assign Recipe to Control Panel................................. Error! Bookmark not defined.
74.13.3 Remove Recipe from Control Panel......................... Error! Bookmark not defined.

75 Resolvers .................. Error! Bookmark not defined.
75.1 Get List of Resolvers...................................................... Error! Bookmark not defined.
75.2 Get Resolver Details
75.3 Add Resolver
75.4 Edit Resolver
75.5 Delete Resolver
76 Resource Pools
76.1 Get List of Resource Pools
76.2 Get Resource Pool Details
76.3 Get Billing Statistics for Resource Pool
76.4 Create Resource Pool
76.5 Edit Resource Pool
76.6 Delete Resource Pool
76.7 Create Resource Pool Data Store
76.8 Edit Resource Pool Data Store
76.9 Delete Resource Pool Data Store
76.10 Get List of Provider Resource Pools
77 Restrictions Sets
77.1 Get List of Restrictions Sets
77.2 Get Restrictions Set Details
77.3 Get List of All Restrictions Resources
77.4 Create Restrictions Set
77.5 Edit Restrictions Set
78 Roles
78.1 Get List of Roles
<table>
<thead>
<tr>
<th>Section</th>
<th>API Function</th>
<th>Description</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>78.2</td>
<td>Get Role Details</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>78.3</td>
<td>Add Role</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>78.4</td>
<td>Edit Role</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>78.5</td>
<td>Delete Role</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>78.6</td>
<td>Edit User Role Assignment</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>78.7</td>
<td>Get List of All Permissions</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>78.8</td>
<td>Clone Role</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79</td>
<td>Smart Servers</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.1</td>
<td>Get List of Smart Servers</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.2</td>
<td>Get Smart Server Details</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.3</td>
<td>Add Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.4</td>
<td>View Encrypted Smart Server Password</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.5</td>
<td>Build Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.6</td>
<td>Edit Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.7</td>
<td>Change Smart Server Owner</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.8</td>
<td>Migrate Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.9</td>
<td>Delete Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.10</td>
<td>Start up Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.11</td>
<td>Reboot Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.12</td>
<td>Reboot Smart Server in Recovery</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.13</td>
<td>Suspend Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.14</td>
<td>Unsuspend Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.15</td>
<td>Unlock Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.16</td>
<td>Shut down Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
<tr>
<td>79.17</td>
<td>Stop Smart Server</td>
<td>Error! Bookmark not defined.</td>
<td></td>
</tr>
</tbody>
</table>
79.18 Open Smart Server Console

79.19 Smart Server Autoscaling

- Get the List of Autoscaling Rules for Smart Server
- Create Autoscaling Rule for SS
- Edit Autoscaling Rule for SS
- Delete Autoscaling Rule

79.20 Smart Server Billing Statistics

79.21 Search Smart Servers by Label

79.22 Get Smart Server CPU Usage Statistics

79.23 Resize Smart Server

79.24 Add/Edit Admin/User Note for Smart Servers

80 Software Licenses

- Get List of Software Licenses
- Get Software License Details
- Add Software License
- Edit Software License
- Delete Software License

81 SSH keys

- Get List of SSH Keys
- Add SSH Key
- Edit SSH Key
- Delete SSH Key

82 Statistics

- Usage Trends Statistics
- Cloud Usage Statistics
83 **Storage Server Backups** ............................................. Error! Bookmark not defined.

83.1 Get All Storage Server Backups Details ............................................. Error! Bookmark not defined.
83.2 Get Normal Storage Server Backups Details ...................................... Error! Bookmark not defined.
83.3 Get Incremental Server Backups Details ............................................. Error! Bookmark not defined.
83.4 Add Backup for Storage Server ....................................................... Error! Bookmark not defined.
83.5 Add/Edit Storage Server Backup Note .............................................. Error! Bookmark not defined.
83.6 Restore Storage Server Backup ........................................................ Error! Bookmark not defined.

84 **System Configuration** .............................................................. Error! Bookmark not defined.

84.1 View System Configuration ............................................................. Error! Bookmark not defined.
84.1.1 Page history ................................................................................ Error! Bookmark not defined.
84.2 Edit System Configuration .............................................................. Error! Bookmark not defined.
84.2.1 Page history ................................................................................ Error! Bookmark not defined.

85 **Service Add-ons** ................................................................. Error! Bookmark not defined.

85.1 Get List of Service Add-ons ............................................................. Error! Bookmark not defined.
85.2 Get Service Add-on Details .............................................................. Error! Bookmark not defined.
85.3 Create Service Add-on ................................................................. Error! Bookmark not defined.
85.4 Edit Service Add-on ................................................................. Error! Bookmark not defined.
85.5 Delete Service Add-on ................................................................. Error! Bookmark not defined.

86 **Service Add-on Events** ............................................................. Error! Bookmark not defined.

86.1 Get List of Service Add-on Events ..................................................... Error! Bookmark not defined.
86.2 Create Service Add-on Recipe Event ................................................ Error! Bookmark not defined.
86.3 Edit Service Add-on Event .............................................................. Error! Bookmark not defined.
86.4 Delete Service Add-on Event ............................................................ Error! Bookmark not defined.

87 **Templates** ................................................................. Error! Bookmark not defined.
### 87. WHITELIST IPS

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
<th>Error! Bookmark not defined.</th>
</tr>
</thead>
<tbody>
<tr>
<td>87.1</td>
<td>Get List of All Templates</td>
<td></td>
</tr>
<tr>
<td>87.2</td>
<td>Get List of System Templates</td>
<td></td>
</tr>
<tr>
<td>87.3</td>
<td>Get List of Own Templates</td>
<td></td>
</tr>
<tr>
<td>87.4</td>
<td>Get List of User Templates</td>
<td></td>
</tr>
<tr>
<td>87.5</td>
<td>Get List of Templates of Particular User</td>
<td></td>
</tr>
<tr>
<td>87.6</td>
<td>Get List of Inactive Templates</td>
<td></td>
</tr>
<tr>
<td>87.7</td>
<td>Get Template Details</td>
<td></td>
</tr>
<tr>
<td>87.8</td>
<td>Make Template Public</td>
<td></td>
</tr>
<tr>
<td>87.9</td>
<td>Delete Template</td>
<td></td>
</tr>
<tr>
<td>87.10</td>
<td>Edit Template</td>
<td></td>
</tr>
<tr>
<td>87.11</td>
<td>Get List of Available for Installation Templates</td>
<td></td>
</tr>
<tr>
<td>87.12</td>
<td>Get List of Available for Update Templates</td>
<td></td>
</tr>
<tr>
<td>87.13</td>
<td>Install Template</td>
<td></td>
</tr>
<tr>
<td>87.14</td>
<td>Update Template</td>
<td></td>
</tr>
<tr>
<td>87.15</td>
<td>Search for Particular Available for Installation Template</td>
<td></td>
</tr>
<tr>
<td>87.16</td>
<td>Search for Particular Available for Upgrade Template</td>
<td></td>
</tr>
</tbody>
</table>

### 88. Template Store

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
<th>Error! Bookmark not defined.</th>
</tr>
</thead>
<tbody>
<tr>
<td>88.1</td>
<td>Get Template Store Details</td>
<td></td>
</tr>
<tr>
<td>88.2</td>
<td>Get Template Group Details</td>
<td></td>
</tr>
<tr>
<td>88.3</td>
<td>Add Template Group</td>
<td></td>
</tr>
<tr>
<td>88.4</td>
<td>Add Child Group</td>
<td></td>
</tr>
<tr>
<td>88.5</td>
<td>Edit Template Group</td>
<td></td>
</tr>
<tr>
<td>88.6</td>
<td>Delete Template Group</td>
<td></td>
</tr>
<tr>
<td>88.7</td>
<td>Get List of Templates Attached to Group</td>
<td></td>
</tr>
<tr>
<td>88.8</td>
<td>Attach Template to Group</td>
<td></td>
</tr>
</tbody>
</table>
88.9 Detach Template from Group ......................................................... Error! Bookmark not defined.
88.10 Change Price for Template in the Group ..................................... Error! Bookmark not defined.

89 Top IOPS Disks .......... Error! Bookmark not defined.
89.1 Get TOP IOPS Disks .................................................................. Error! Bookmark not defined.

90 Transactions .......... Error! Bookmark not defined.
90.1 Get List of Transactions .......................................................... Error! Bookmark not defined.
90.2 Get List of VS Transactions without Log Output ....................... Error! Bookmark not defined.
90.3 Get List of Transactions with Log Output .................................. Error! Bookmark not defined.
90.4 Get Transaction Details .......................................................... Error! Bookmark not defined.
90.5 Get Transaction Details without Log Output .............................. Error! Bookmark not defined.

91 User Additional Fields .. Error! Bookmark not defined.
91.1 Get List of Additional Fields .................................................. Error! Bookmark not defined.
91.2 Get Additional Field Details .................................................. Error! Bookmark not defined.
91.3 Add Additional Field ............................................................ Error! Bookmark not defined.
91.4 Edit Additional Field ............................................................. Error! Bookmark not defined.
91.5 Delete Additional Field .......................................................... Error! Bookmark not defined.
91.6 Search User by Additional Field Parameter ............................. Error! Bookmark not defined.

92 User Groups .............. Error! Bookmark not defined.
92.1 Get List of User Groups .......................................................... Error! Bookmark not defined.
92.2 Get User Group Details .......................................................... Error! Bookmark not defined.
92.3 Add User Group ................................................................. Error! Bookmark not defined.
92.4 Edit User Group ................................................................. Error! Bookmark not defined.
92.5 Delete User Group ............................................................... Error! Bookmark not defined.
93 Users

93.1 Get List of Users

93.2 Get User Details

93.3 Check Login/E-mail Availability

93.4 Add User

93.5 Edit User

93.6 Suspend User

93.7 Activate User

93.8 Unlock User

93.9 Delete User

93.10 Drop Sessions

93.11 Get User Last Access Log

93.12 Get User Statistics

93.13 Get User’s Statistics for Particular Period

93.14 Get User Billing Statistics

93.15 Get List of User Monthly Bills

93.16 Get List of User Payments

93.17 Add Payment

93.18 Edit Payment

93.19 Delete Payment

93.20 Get List of User VSS

93.21 Get List of Compute resources Used by Users’ VSS

93.22 Get List of User Backups

93.23 Get List of User Data Store Zones
93.24 Get List of User Limits .................................................. Error! Bookmark not defined.
93.25 Get List of User Network Zones ...................................... Error! Bookmark not defined.
93.26 Search User Backups ...................................................... Error! Bookmark not defined.
93.27 Generate API Key .......................................................... Error! Bookmark not defined.
93.28 Search Users by Name .................................................... Error! Bookmark not defined.
93.29 Get List of User’s YubiKeys .............................................. Error! Bookmark not defined.
93.30 Add Yubikey to User ...................................................... Error! Bookmark not defined.
93.31 Delete User Yubikey ...................................................... Error! Bookmark not defined.

94 Users with Config Problems ...............Error! Bookmark not defined.
94.1 Get the list of users without roles ...................................... Error! Bookmark not defined.
94.2 Get the list of users without timezones ................................ Error! Bookmark not defined.
94.3 Get the list of users without user groups .............................. Error! Bookmark not defined.

95 vApps .................Error! Bookmark not defined.
95.1 Get List of vApps .......................................................... Error! Bookmark not defined.
95.2 Get List of vApp VSs ...................................................... Error! Bookmark not defined.
95.3 Get vApp Details ........................................................... Error! Bookmark not defined.
95.4 Recompose vApp .......................................................... Error! Bookmark not defined.
95.5 Edit vApp ................................................................. Error! Bookmark not defined.
95.6 Start vApp ................................................................. Error! Bookmark not defined.
95.7 Stop vApp ................................................................. Error! Bookmark not defined.
95.8 Delete vApp ............................................................... Error! Bookmark not defined.
95.9 Create vApp ............................................................... Error! Bookmark not defined.
95.10 Compose vApp ........................................................... Error! Bookmark not defined.
95.11 Create vApp with Guest Customization and Recipes .......... Error! Bookmark not defined.

96 vApp Networks ............ Error! Bookmark not defined.
96.1 Get List of vApp Networks........................................... Error! Bookmark not defined.
96.2 Get vApp Network Details............................................. Error! Bookmark not defined.
96.3 Get List of IP Addresses Assigned to vApp Network ......... Error! Bookmark not defined.

97 vCloud External Networks Error! Bookmark not defined.
97.1 Get List of vCloud External Networks............................... Error! Bookmark not defined.
97.2 Get vCloud External Network Details................................. Error! Bookmark not defined.
97.3 Get List of IP Addresses Assigned to vCloud External Network .... Error! Bookmark not defined.

98 vCloud Firewall Rules ... Error! Bookmark not defined.
98.1 Get List of vCloud Firewall Rules...................................... Error! Bookmark not defined.
98.2 Get Firewall Rule Details.............................................. Error! Bookmark not defined.
98.3 Create vCloud Firewall Rule ....................................... Error! Bookmark not defined.
98.4 Edit vCloud Firewall Rules........................................... Error! Bookmark not defined.
98.5 Delete vCloud Firewall Rule.......................................... Error! Bookmark not defined.

99 vCloud Network Interfaces ........ Error! Bookmark not defined.
99.1 Get List of vCloud Network Interfaces............................. Error! Bookmark not defined.
99.2 Get vCloud Network Interface Details.............................. Error! Bookmark not defined.
99.3 Create vCloud Network Interface................................. Error! Bookmark not defined.
99.4 Edit vCloud Network Interface.................................. Error! Bookmark not defined.
99.5 Delete vCloud Network Interface................................ Error! Bookmark not defined.
100 vCloud Organization Networks .......... Error! Bookmark not defined.

100.1 Get List of vCloud Organization Networks ......................... Error! Bookmark not defined.
100.2 Get vCloud Organization Network Details .............................. Error! Bookmark not defined.
100.3 Create vCloud Organization Network .................................. Error! Bookmark not defined.
100.4 Edit vCloud Organization Network ...................................... Error! Bookmark not defined.
100.5 Delete vCloud Organization Network .................................. Error! Bookmark not defined.

101 vCloud VS Snapshots .......... Error! Bookmark not defined.

101.1 Get List of vCloud VS Snapshots ...................................... Error! Bookmark not defined.
101.2 Create vCloud VS Snapshot .............................................. Error! Bookmark not defined.
101.3 Build vCloud VS Snapshot ................................................ Error! Bookmark not defined.
101.4 Restore vCloud VS Snapshot .............................................. Error! Bookmark not defined.
101.5 Delete vCloud VS Snapshot .............................................. Error! Bookmark not defined.

102 Version .......................... Error! Bookmark not defined.

103 Virtual Servers .............. Error! Bookmark not defined.

103.1 Get List of VSs .......................................................... Error! Bookmark not defined.
103.1.1 Page History: ................................................................. Error! Bookmark not defined.
103.2 Get VS Details .............................................................. Error! Bookmark not defined.
103.2.1 Page History: ................................................................. Error! Bookmark not defined.
103.3 Get Statuses for all Virtual Servers .................................. Error! Bookmark not defined.
103.4 Get Virtual Server Status ................................................ Error! Bookmark not defined.
103.5 Add VS ........................................................................ Error! Bookmark not defined.
103.5.1 Page history ................................................................. Error! Bookmark not defined.
103.6 Add Instance Package VS ................................................ Error! Bookmark not defined.
103.7 Add VMware VS ............................................................ Error! Bookmark not defined.
103.7.1 Page history ................................................................. Error! Bookmark not defined.
<table>
<thead>
<tr>
<th>Page</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>103.8</td>
<td>View Encrypted VS Password</td>
</tr>
<tr>
<td>103.9</td>
<td>Build or Rebuild VS</td>
</tr>
<tr>
<td>103.10</td>
<td>Edit VS</td>
</tr>
<tr>
<td>103.11</td>
<td>Change VS Owner</td>
</tr>
<tr>
<td>103.12</td>
<td>Reset VS Root Password</td>
</tr>
<tr>
<td>103.13</td>
<td>Set SSH Keys</td>
</tr>
<tr>
<td>103.14</td>
<td>Migrate VS</td>
</tr>
<tr>
<td>103.15</td>
<td>Set VIP Status for VS</td>
</tr>
<tr>
<td>103.16</td>
<td>Delete VS</td>
</tr>
<tr>
<td>103.17</td>
<td>Start up VS</td>
</tr>
<tr>
<td>103.18</td>
<td>Segregate VS</td>
</tr>
<tr>
<td>103.19</td>
<td>Desegregate VS</td>
</tr>
<tr>
<td>103.20</td>
<td>Reboot VS</td>
</tr>
<tr>
<td>103.21</td>
<td>Enable Accelerator for Virtual Server</td>
</tr>
<tr>
<td>103.22</td>
<td>Disable Accelerator for Virtual Server</td>
</tr>
<tr>
<td>103.23</td>
<td>Purge File(s)</td>
</tr>
<tr>
<td>103.24</td>
<td>Purge All Content</td>
</tr>
<tr>
<td>103.25</td>
<td>Reboot VS in Recovery</td>
</tr>
<tr>
<td>103.26</td>
<td>Reboot VS from ISO</td>
</tr>
<tr>
<td>103.27</td>
<td>Boot VS from ISO</td>
</tr>
<tr>
<td>103.28</td>
<td>Suspend VS</td>
</tr>
<tr>
<td>103.29</td>
<td>Unlock VS</td>
</tr>
<tr>
<td>103.30</td>
<td>Unsuspend VS</td>
</tr>
<tr>
<td>103.31</td>
<td>Shut down VS</td>
</tr>
<tr>
<td>103.32</td>
<td>Stop VS</td>
</tr>
</tbody>
</table>
103.33  Open VS Console ................................................................. Error! Bookmark not defined.

103.34  VS Autoscaling ........................................................................ Error! Bookmark not defined.
103.34.1 Enable Autoscaling for VS ................................................... Error! Bookmark not defined.
103.34.2 Get the List of Autoscaling Rules for VS .............................. Error! Bookmark not defined.
103.34.3 Create Autoscaling Rule for VS .......................................... Error! Bookmark not defined.
103.34.4 Edit Autoscaling Rule for VS ............................................. Error! Bookmark not defined.
103.34.5 Delete Autoscaling Rule ..................................................... Error! Bookmark not defined.
103.34.6 Disable Autoscaling for VS ................................................. Error! Bookmark not defined.

103.35  VS Billing Statistics ................................................................. Error! Bookmark not defined.
103.35.1 Page history ........................................................................ Error! Bookmark not defined.

103.36  Search VS by Label ................................................................. Error! Bookmark not defined.

103.37  Get VS CPU Usage Statistics ................................................ Error! Bookmark not defined.

103.38  Add/Edit Admin/User Note for Virtual Server ........................ Error! Bookmark not defined.
103.38.1 Add/Edit User Note ............................................................. Error! Bookmark not defined.

103.39  Enable Booting from CD for ISO Virtual Server ..................... Error! Bookmark not defined.

103.40  Disable Booting from CD for ISO Virtual Server ..................... Error! Bookmark not defined.

103.41  Add Media File to vCloud VS ................................................ Error! Bookmark not defined.

103.42  Remove Media File from vCloud VS ....................................... Error! Bookmark not defined.

103.43  View vCloud VS Guest Customization .................................. Error! Bookmark not defined.

103.44  Get List of Service Add-ons Assigned to VS .......................... Error! Bookmark not defined.

103.45  Assign Service Add-on to VS ................................................ Error! Bookmark not defined.

103.46  Unassign Service Add-on from VS ........................................ Error! Bookmark not defined.

103.47  Use VS as Gateway ............................................................... Error! Bookmark not defined.

104  Whitelisted IPs .......... Error! Bookmark not defined.

104.1  Get List of Whitelist IPs .......................................................... Error! Bookmark not defined.

104.2  Get Whitelist IP Details .......................................................... Error! Bookmark not defined.

104.3  Add Whitelisted IP ................................................................. Error! Bookmark not defined.

104.4  Edit Whitelisted IP ................................................................. Error! Bookmark not defined.
104.5  Delete Whitelisted IP ................................................................. Error! Bookmark not defined.
The API enables cloud integration with third party applications. You can manage every aspect of your cloud through the API. 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: updated_at value is changed in PUT requests even if the request fails.

**DELETE** - used for object deletion
## 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>OK \text{ The request completed successfully}</td>
</tr>
<tr>
<td>204</td>
<td>No content \text{ The request completed successfully. The 204 status is returned on DELETE and PUT requests}</td>
</tr>
<tr>
<td>201</td>
<td>Scheduled \text{ The request has been accepted and scheduled for processing}</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden \text{ The request is correct, but could not be processed.}</td>
</tr>
<tr>
<td>404</td>
<td>Not Found \text{ 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</td>
<td>Unprocessable Entity \text{ The sent parameters are erroneous.}</td>
</tr>
<tr>
<td>500</td>
<td>Internal Server Error \text{ An error occurred. Please contact support.}</td>
</tr>
<tr>
<td>503</td>
<td>Service Unavailable \text{ 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 <em>username:password</em> 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>* (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>
</tbody>
</table>

**Code block** indicates console requests and response examples.

| 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 completion. | The maximum length of a Mount Point is 256 characters. |
| warning | A warning message informs you of something you should not do or be cautious. | You won't be able to restore a VS after deleting it. |
| limit_type | The element showing new parameters added in the latest release of API. | *limit_type* – hourly or monthly limit type set for the resource |
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 you create a VS on behalf of another user?
A: No. It is possible to switch VS owners, however. 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 ONAPP 5.3

Added
- Added new sections for CDN reports:
  - CDN Overview Report
  - CDN Cache Statistics Report
  - CDN Status Codes Report
  - CDN Bandwidth Statistics Report
- Added new docs for Service Add-ons:
  - Service Add-ons
  - Service Add-on Events chapter
  - Get List of Service Add-ons Assigned to VS
  - Assign Service Add-on to VS
  - Unassign Service Add-on from VS
  - Add Limits for Service Add-on Groups
- Added Use VS as Gateway section
- Added Unlock OVA section
- Added Get Resource Difference Details section
- Added Add File to Media Library section

Updated
- Updated Container Servers chapter: added new API requests
- Updated Edit LVM Data Store section: added new auto_healing parameter
- Updated Add Data Store Zone, Add Backup Server Zone and Add Network Zone sections: added new server_type parameter
- Updated Add Compute Zone section: added vpc option for the server_type parameter
- Updated the following sections with information on zone types:
  - Attach Compute Resource to Compute Zone
  - Add Data Store Join to Compute Zone
  - Add Network Join to Compute Zone
  - Add Backup Server to Compute Zone
  - Add Data Store Join to Compute Resource
  - Add Network Join to Compute Resource
- Add Backup Server to Compute Resource
- Assign Backup Server to Backup Server Zone
- Attach Network to Network Zone
- Attach Data Store to Data Store Zone

- Updated Recompose vApp section: added parameters on guest customization and recipes
- Added information on service add-ons and limits for service add-on groups to the following documents:
  - Billing Plans
  - Get Base Resources Details
  - VS Billing Statistics
- Updated Get VS Details section: added new template_version parameter
- Updated Get VS Log Item Details, Get List of Log Items and Get List of VS Log Items sections: added new resource_diff_id parameter
- Updated Get List of DNS Zone Records and Edit DNS Records sections: added new PTR array of records with their parameters
- Updated Segregate VS and Desegregate VS sections: removed deprecated POST/virtual_machines/:virtual_machine_id/strict_vm method
- Updated Add VS section: recipe_ids and custom_variables parameters replaced with recipe_joins_attributes and custom_variables_attributes parameters respectively
- Updated the following sections with the announcement that the requests will be deprecated or changed in OnApp 5.4 version:
  - Add IP Address Record
  - Assign IP Address to User
  - Get List of Network IP Addresses
  - Assign IP Address Join to VS
  - Delete IP Address Join
7 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
```bash
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
```bash
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.
8  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 Output example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<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>
```

Explanation of the data returned:

- **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 VSs, which are running on a compute resource, but are not in the DB
9 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

9.1 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
xml</label><hostname>zaza</hostname><hypervisor_group_id>6</hypervi
sor_group_id><hypervisor_id>2</hypervisor_id><memory>384</memory><c
pus>1</cpus><cpu_shares>1</cpu_shares><data_store_group_primary_id>
1</data_store_group_primary_id><primary_disk_size>5</primary_disk_s
ize><data_store_group_swap_id>1</data_store_group_swap_id><swap_dis
k_size>1</swap_disk_size><primary_network_group_id>16</primary_net
work_group_id><required_ip_address_assignment>1</required_ip_address
_assignment><rate_limit>0</rate_limit><required_virtual_machine_bui
ld>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", "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",
"required_ip_address_assignment":"1", "rate_limit":"0",
"required_virtual_machine_build":"1"}}' --url
http://onapp.test/application_servers.xml

XML Output example

<application_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>
  <booted type="boolean">false</booted>
  <built type="boolean">false</built>
  <cores_per_socket type="integer">0</cores_per_socket>
  <cpu_shares type="integer">1</cpu_shares>
  <cpu_sockets nil="true"/>
  <cpu_threads nil="true"/>
  <cpu_units type="integer">10</cpu_units>
  <cpus type="integer">1</cpus>
  <created_at type="datetime">2015-06-23T15:33:36+03:00</created_at>
  <customer_network_id nil="true"/>
  <deleted_at nil="true"/>
  <edge_server_type nil="true"/>
  <enable_autoscale nil="true"/>
<enable_monitis nil="true"/>
<firewall_notrack type="boolean">false</firewall_notrack>
<hostname>zaza</hostname>
<hot_add_cpu nil="true"/>
<hot_add_memory nil="true"/>
<hypervisor_id type="integer">2</hypervisor_id>
<id type="integer">411</id>
<identifier>tb6s00085zqig2</identifier>
<initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
<instance_package_id nil="true"/>
<iso_id nil="true"/>
<label>zaza xml</label>

<local_remote_access_ip_address>10.0.24.32</local_remote_access_ip_address>
<local_remote_access_port nil="true"/>
<locked type="boolean">true</locked>
<memory type="integer">384</memory>
.getMin_disk_size type="integer">5</min_disk_size>
<note nil="true"/>
<operating_system>linux</operating_system>
<operating_system_distro>rhel</operating_system_distro>
<preferred_hvs type="array"/>
<recovery_mode nil="true"/>
<remote_access_password>ReO44gmSyF92</remote_access_password>
<service_password nil="true"/>
<state>building</state>
<storage_server_type nil="true"/>
<strict_virtual_machine_id nil="true"/>
<suspended type="boolean">false</suspended>
<template_id type="integer">155</template_id>
<template_label>CentOS 6.6 x64 ApplicationServer</template_label>
<time_zone nil="true"/>
<updated_at type="datetime">2015-06-23T15:33:41+03:00</updated_at>
<br>user_id type="integer">1</user_id>
<vip nil="true"/>
<xen_id nil="true"/>
<ip_addresses type="array">
<ip_address>
<address>69.168.237.56</address>
<broadcast>69.168.237.255</broadcast>
<created_at type="datetime">2015-05-14T12:19:08+03:00</created_at>
<customer_network_id nil="true"/>
<disallowed_primary type="boolean">false</disallowed_primary>
<gateway>69.168.237.1</gateway>
<hypervisor_id nil="true"/>
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
- **cpu_shares** - 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 user billing plan.
- **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 is 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
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 billing plan)
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][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 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

9.1.1 Page History

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
10 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.

10.1 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

```bash
```

JSON Request example

```bash
```

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

10.2 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

<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>
......</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

### 10.3 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
10.4 Install Application

To install application:
POST /application_servers/:application_server_id/applications.xml
POST /application_servers/:application_server_id/applications.json

**XML Request example**
curl -i -X POST -u user:userpass
'&lt;application&gt;&lt;script_id&gt;1&lt;/script_id&gt;&lt;softdirectory&gt;XML_Zikula&lt;/softdirectory&gt;&lt;admin_username&gt;admin&lt;/admin_username&gt;&lt;admin_pass&gt;pass&lt;/admin_pass&gt;&lt;admin_email&gt;user@onapp.com&lt;/admin_email&gt;&lt;/application&gt;'

**JSON Request example**
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/applications.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"application": {"script_id": "1", "softdirectory": "Zikula1000", "admin_username": "admin", "admin_pass": "pass", "admin_email": "user@onapp.com"}}'

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.

10.5 Back Up Application

To back up an application:
POST /application_servers/:application_server_id/applications/:id/backup.xml
POST /application_servers/:application_server_id/applications/:id/backup.json
**XML Request example**


'<?xml version="1.0" encoding="UTF-8"?>
<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**


`{"backup":
{"application_id": "1_23728", "backup_directory": "1", "backup_database": "1", "note": "json backup"} }`
OnApp 5.3 API Guide    v2

WHITELIST IPS

EDIT WHITELISTED IP

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>
</backups>

10.7 Get List of All Application Backups

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
<backup>
<application_id>123_70977</application_id>
<application_type>CodeIgniter</application_type>
<backup_note>Code Igniter Backup #1</backup_note>
<identifier>7d0c93305dc816282e17e432903e33eb</identifier>
<software_version>3.0.0</software_version>
<size>1.369 MB</size>
</backup>

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

### 10.8 Restore Application Backup

To restore application backup:

**POST**
```
/post
```

**XML Request example**
```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
```

```
http://onapp.test/application_servers/:application_server_id/applications/backups/:identifier/restore.xml
```

```
'd '<backup>
<restore_directory>1</restore_directory>
<restore_database>1</restore_database>
</backup>'
```

**JSON Request example**
```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
```

```
http://onapp.test/application_servers/:application_server_id/applications/backups/:identifier/restore.json
```

```
d
```
"backup": {"restore_directory": "1", "restore_database": "1"} }

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 ApplicationBackups request.

10.9 Remove Application Backup

To remove application backup:

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
```plaintext
```

JSON Request example
```plaintext
```

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

10.10 System Applications

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

10.10.1 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/:application_server_id/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/application_server_id/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>
  <!-- more entries -->
</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

### 10.10.2 Install System Application

To install a system application:

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**

```
curl -i -X PUT -u user:userpass
http://onapp.test/application_servers/:application_server_id/system
/apps/system_app_id/install.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/:application_server_id/system
```

WHITELIST IPS - EDIT WHITELISTED IP 71
_apps/system_app_id/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.

10.10.3 Switch PHP Version

To switch a PHP version:
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  

JSON Request example

curl -i -X PUT -u user:userpass  

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

10.10.4 Uninstall System Application

To uninstall a system application:
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  

JSON Request example

curl -i -X PUT -u user:userpass  
http://onapp.test/application_servers/:application_server_id/system_apps/system_app_id/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
You can get system application ID with the Get List of System Applications API request.

10.11 Domains

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

10.11.1 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


JSON Request example


XML Output example

<domains type="array">
  <domain>
    <identifier>3dd0d7734983e2db8ba7677bd5b11a70</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

### 10.11.2 Create Domain

To create a domain, use the following request:

```plaintext
POST /application_servers/:application_server_id/domains.xml
POST /application_servers/:application_server_id/domains.json
```

There are several options for domain creation:
- Addon domain creation with custom path
- Domain addition to an existing application
- Creation of parked domain

#### 10.11.2.1 Addon domain creation with custom path

**XML Request example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/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/:application_server_id/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

#### 10.11.2.2 Domain addition to an existing application

**XML Request example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/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/:application_server_id/domains.json
-H 'Accept: application/json' -H 'Content-type: application/json'
-d '{"domain": {"path": "ZikulaTest","domain": "addon.com"}}'
```
'<!--domain-->existed.com</domain><application_id>None</application_id></domain>''

**JSON Request example**

curl -i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/domains.json '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

**10.11.2.3 Creation of parked domain**

**XML Request example**

curl -i -X POST -u user:userpass
'<!--domain-->parked.com</domain><application_id>Null</application_id></domain>''

**JSON Request example**

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

Where:

domain - the name of the domain
10.11.3 Delete Domain

To delete a domain:
DELETE /application_servers/:application_server_id/domains/:domain_identifier.xml
DELETE /application_servers/:application_server_id/domains/:domain_identifier.json

**XML Request example**

**JSON Request example**

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

10.12 FTP Users

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

10.12.1 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**
curl -i -X GET -u user:userpass -H 'Accept: application/json' -H
WHITELIST IPS

WHITELISTED IP

'Content-type: application/json' --url
http://onapp.test/application_servers/:application_server_id/ftp_users.json'

XML Output example
<?xml version="1.0" encoding="UTF-8"?>
<ftp_users type="array">
    <ftp_user>
        <identifier>eb9baa2f057e4535a17862e5f2ab5</identifier>
        <login>onapp</login>
        <path>/home/onapp</path>
        <usage type="integer">0</usage>
    </ftp_user>
    <ftp_user>
        <identifier>0fd3345d0c4b4c20b46f8ee33f52ba75</identifier>
        <login>test_onapp.test</login>
        <path>/home/onapp/www/test</path>
        <usage type="integer">0</usage>
    </ftp_user>
    <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)

10.12.2 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
'"ftp_user":<password>1234</password><password_confirmation>1234</password_confirmation><login>login364</login><path>www/usr2</path>"

JSON Request example

curl -i -X POST -u user:userpass
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

### 10.12.3 Delete FTP User

To delete an FTP user:

```
DELETE
/application_servers/:application_server_id/ftp_users/:ftp_user_identifier.xml
```

```
DELETE
/application_servers/:application_server_id/ftp_users/:ftp_user_identifier.json
```

**XML Request example**

```
```

**JSON Request example**

```
```

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**

### 10.13 Databases

This section contains the API requests you can apply to manage databases available for your Application Server.
10.13.1 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**


**JSON Request example**


**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<databases type="array">
  <database>
    <db>Test1</db>
  </database>
  <database>
    <db>cmfsfs</db>
  </database>
</databases>
```

Where:

db - the name of the database

10.13.2 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

**JSON Request example**

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

To delete a database:

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 -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
'http://onapp.test/application_servers/:application_server_id/databases/:db.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/:application_server_id/databases/:db.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

10.13.4  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

```
curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url
'http://onapp.test/application_servers/:application_server_id/database_users.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/:application_server_id/database_users.json'
```
XML Output example
<?xml version="1.0" encoding="UTF-8"?>
<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

10.13.5 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
<?xml version="1.0" encoding="UTF-8"?>
<database_users type="array">
    <database_user>
        <privileges>
            <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>
        </privileges>
    </database_user>
</database_users>
WHITELIST IPS

<INDEX>false</INDEX>
<CREATE_ROUTINE>false</CREATE_ROUTINE>
<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
LOCK_TABLES
REFERENCES
CREATE_ROUTINE
CREATE_VIEW
SHOW_VIEW
TRIGGER

name - the name of the database user

10.13.6 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
'<database_user><name>user</name><password>pass</password></database_user>'

JSON Request example
WHITELISTED IPS

### WHITELIST IPS - EDIT WHITELISTED IP

**curl**

```bash
-i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/databases/:db/assign_user.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<database><db_user>user</db_user><host>localhost</host><prilist><SELECT>true</SELECT></prilist></database>'
```

**Where:**

- **name** - the name of the database user
  - The length of name should not exceed 11 characters.
- **password** - the password for the database user

### 10.13.7 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
```

**XML Request example**

```bash
-i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/databases/:db/assign_user.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<database><db_user>user</db_user><host>localhost</host><prilist><SELECT>true</SELECT></prilist></database>'
```

**JSON Request example**

```bash
-i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/databases/:db/assign_user.json
-H 'Accept: application/json' -H 'Content-type: application/json' -d '{"database": {"db_user": "user", "host": "localhost", "prilist": {"SELECT":true}}}'
```

**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

10.13.8  Update Database User Privileges

To update a database user privileges, 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/:application_server_id/database_users/:name/privileges.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<database_user><db_name>label</db_name><host>localhost</host><privlist><SELECT>true</SELECT></privlist></database_user>'`

JSON Request example
`curl -i -X PUT -u user:userpass http://onapp.test/application_servers/:application_server_id/database_users/:name/privileges.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"database_user": {"db_name": "label", "host": "localhost", "privlist":{"SELECT":true}}}'}

Where:
db_name - the name of the database
privlist - 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
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

10.13.9 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
http://onapp.test/application_servers/:application_server_id/database_users/:name/change_password.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"database_user": {"password": "newpass"}}'

Where:
password - the password for the database user

10.13.10 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
WHITELIST IPS

EDIT

WHITELISTED IP

86

'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/:application_server_id/database_users/:name/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

10.13.11 Delete Database User

To delete a database user:

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

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

10.14 Email Accounts

This section contains the API requests you can apply to manage email accounts for domains.
10.14.1 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**

```
curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H
  'Content-type: application/xml' --url
  http://onapp.test/application_servers/:application_server_id/email_accounts.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/:application_server_id/email_accounts.json
```

**XML Output example**

```xml
<?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>8fbbfca4101c61b5b8c5d0157353dc8b</identifier>
    <space></space>
    <user>test2@109.123.91.19</user>
  </email_account>
</email_accounts>
```

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

10.14.2 Get List of Email Accounts for Specific Domain

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**

```
curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H
  'Content-type: application/xml' --url
  http://onapp.test/application_servers/:application_server_id/email_accounts?domain=example.com.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/:application_server_id/email_accounts?domain=example.com.json
```
WHITELIST IPS

EDIT WHITELISTED IP

accounts?domain=example.com.xml'

JSON Request example

```
```

XML Output example

```xml
<?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>
```

Where:

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

### 10.14.3 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

```
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/email_accounts.xml 'Accept: application/xml' -H 'Content-type: application/xml' -d
  '<email_account><password>1234</password><password_confirmation>1234</password_confirmation><user>login364</user><domain>example.com</domain></email_account>'
```

JSON Request example

```
curl -i -X POST -u user:userpass
http://onapp.test/application_servers/:application_server_id/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

10.14.4 Delete Email Account for Default Domain

To delete an email account:
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

10.14.5 Delete Email Account for Specific Domain

To delete an email account for a specific domain:
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

```
curl -i -X DELETE -u user:userpass --url
http://onapp.test/application_servers/:application_server_id/email_accounts/:email_account_identifier.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d '<domain_name>existed.com</domain_name>'
```

JSON Request example

```
curl -i -X DELETE -u user:userpass --url
http://onapp.test/application_servers/:application_server_id/email_accounts/:email_account_identifier.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"domain_name": "existed.com"}''
```

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

10.15 Services

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

10.15.1 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

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

JSON Request example
curl -i -X GET -u user:pass -H 'Accept: application/json' -H 'Content-type: application/json' --url http://onapp.test/application_servers/:application_server_id/services.json

XML Output example
<?xml version="1.0" encoding="UTF-8"?><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

10.15.2 Start/Stop/Restart 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

10.15.2.1 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
```
curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url http://onapp.test/application_servers/:application_server_id/services/:service_id/start.xml'
```

JSON Request example
```
curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' --url http://onapp.test/application_servers/:application_server_id/services/:service_id/start.json'
```

### 10.15.2.2 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
```
curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url http://onapp.test/application_servers/:application_server_id/services/:service_id/stop.xml'
```

JSON Request example
```
curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' --url http://onapp.test/application_servers/:application_server_id/services/:service_id/stop.json'
```

### 10.15.2.3 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
```
curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' --url http://onapp.test/application_servers/:application_server_id/services/:service_id/restart.json'
```
11 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.

11.1 Get List of Assets

Use the following API call to view the list of assets:

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

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

XML Request example


JSON Request example


XML Output example

<?xml version="1.0" encoding="UTF-8"?>
<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

11.2 Get Asset Details

Use the following API call to view the list of assets:
GET /settings/assets/:asset_mac_address.xml
GET /settings/assets/:asset_mac_address.json

**XML Request example**

curl -i -u user:userpass
http://onapp.test/settings/assets/:asset_mac_address.xml -H
   'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request example**

curl -i -u user:userpass
http://onapp.test/settings/assets/:asset_mac_address.json -H
   'Accept: application/json' -H 'Content-type: application/json'

**XML Response example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<asset>
   <mac nil="true"/>
   <disks type="array">
      <disk>
         <name>disk1name1</name>
         <scsi>disk1scsi</scsi>
      </disk>
      <disk>
         <name>disk3name3</name>
         <scsi>disk3scsi</scsi>
      </disk>
   </disks>
   <nics type="array">
      <nic>
         <name>nic1name1</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

### 11.3 Get List of Unassigned Assets

Use the following API call to view the list of assets:

```
GET hypervisors/not_grouped.xml
GET hypervisors/not_grouped.json
```

**XML Request example**

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

**JSON Request example**

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

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<hypervisors type="array">
  <hypervisor>
    <backup type="boolean">false</backup>
    <backup_ip_address/>
    <blocked type="boolean">true</blocked>
    <built type="boolean">false</built>
    <called_in_at nil="true"/>
    <connection_options nil="true"/>
    <cpu_idle type="integer">0</cpu_idle>
    <cpu_mhz nil="true"/><cpus nil="true"/>
    <created_at type="datetime">2013-06-10T12:09:48+00:00</created_at>
    <custom_config nil="true"/>
    <disable_failover type="boolean">false</disable_failover>
    <disk_pcis nil="true"/>
    <distro nil="true"/>
    <enabled type="boolean">true</enabled>
  </hypervisor>
</hypervisors>
```
<failure_count type="integer">0</failure_count>
<format_disks type="boolean">false</format_disks>
<free_mem type="integer">0</free_mem>
<host nil="true"/>
<hypervisor_group_id nil="true"/>
<hypervisor_type>kvm</hypervisor_type>
<id type="integer">7</id>
<ip_address>109.123.105.132</ip_address>
<label>KVM C5 HV1</label>
<list_of_logical_volumes/>
<list_of_volume_groups/>
<list_of_zombie_domains nil="true"/>
<locked type="boolean">false</locked>
<mac nil="true"/>
<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"/>
<server_type>virtual</server_type>
<spare type="boolean">false</spare>
<storage_channel>224.3.28.1</storage_channel>
<threads_per_core nil="true"/>
<total_mem nil="true"/>
<total_zombie_mem nil="true"/>
<updated_at type="datetime">2013-06-10T12:09:48+00:00</updated_at>
<uptime nil="true"/>
<vmmware_total_cpu_cores type="integer">0</vmmware_total_cpu_cores>
<total_cpus type="integer">0</total_cpus>
<free_memory type="integer">0</free_memory>
<used_cpu_resources type="integer">0</used_cpu_resources>
<total_memory type="integer">0</total_memory>
<cpu_cores type="integer">0</cpu_cores>
<free_disk_space><onapp-fv4zl7t2h5wbeq type="integer">184</onapp-fv4zl7t2h5wbeq></free_disk_space>
<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>
</hypervisors>

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

`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

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

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
12 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.

12.1 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.

12.1.1 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

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

XML Output example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<autobackup_templates type="array">
<autobackup_template>
<duration type="integer">1</duration>
<created_at type="datetime">2011-07-14T15:01:38Z</created_at>
<updated_at type="datetime">2011-07-28T11:49:52Z</updated_at>
<period>days</period>
<id type="integer">1</id>
<enabled type="boolean">true</enabled>
</autobackup_template>
<autobackup_template>
<duration type="integer">1</duration>
<created_at type="datetime">2011-07-14T15:01:38Z</created_at>
<updated_at type="datetime">2011-07-28T11:50:21Z</updated_at>
```
**WHITELIST IPS - EDIT WHITELISTED IP**

102

---

12.1.2 Get Auto-backup Preset Details

This method will output the details for a particular auto-backup preset.

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

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?><autobackup_template>
  <duration type="integer">1</duration> <created_at type="datetime">2011-01-06T10:49:43Z</created_at>
  <period>days</period> <updated_at type="datetime">2011-01-06T10:49:43Z</updated_at> <enabled type="boolean">true</enabled>
  <id type="integer">1</id></autobackup_template>
```

Where:

- **duration** - edit the number specifying how often a backup should be taken
- **created_at** - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- **period** - specifies the time period (days, weeks, months, or years)
- **updated_at** - the date when the auto-backup preset was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **enabled** - true if the auto-backup preset is enabled, otherwise false.
- **id** - the ID of the auto-backup preset

12.1.3 Edit Auto-backup Preset

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

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

**XML Request example**

```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' ...
```
application/xml' -u user:userpass -d'<?xml version="1.0" encoding="UTF-8"?><autobackup_template><duration>5</duration><period>days</period><enabled>false</enabled></autobackup_template>' --url http://onapp.test/settings/autobackup_presets/:id.xml'

**JSON Request example**
```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d'{"autobackup_template":{"duration":"5","period":"weeks","enabled":false}}' --url http://onapp.test/settings/autobackup_presets/:id.json'
```

You can edit the following parameters:
- **duration** - edit the number specifying how often a backup should be taken
- **enabled** - set true if auto-backup preset is enabled, otherwise set false

Every autobackup_preset_id has its defined period (either days, or weeks, or months, or years), which cannot be altered.

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.

## 12.2 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.
12.2.1 Enable auto-backups for VS

To enable incremental auto-backups for a Virtual Server:

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/:id/autobackup_enable.xml -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/:id/autobackup_enable.json -u
user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

12.2.2 Disable auto-backups for VS

To disable incremental auto-backups for a Virtual Server:

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

XML Request example
```bash
curl -i -X POST
http://onapp.test/virtual_machines/:id/autobackup_disable.xml -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/:id/autobackup_disable.json -u
user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
```

12.2.3 Enable Auto-backups for Disk

You can enable auto-backups for a disk using the following methods:

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

XML Request example
```bash
curl -i -X POST
http://onapp.test/settings/disks/:disk_id/autobackup_enable.xml -u
user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request example
```bash
curl -i -X POST
http://onapp.test/settings/disks/:disk_id/autobackup_enable.json -u
```
12.2.4 Disable Auto-backups for Disk

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

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

XML Request example

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

JSON Request example

curl -i -X POST
http://onapp.test/settings/disks/:disk_id/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

12.3 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.

12.3.1 Get List of All Schedules

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

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

JSON Request example


XML Output example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<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>
    <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>
</schedules>
```

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][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** - 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

**UserID** - 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

### 12.3.1.1 Page History

**v3.2**

Added the following parameters:

- *rotation_period*

### 12.3.2 Get Schedule Details

Use this method to get details for a particular disk backup schedule:

GET /settings/schedules/:id.xml

GET /settings/schedules/:id.json

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**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<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>
</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
Added the following parameters:

`rotation_period`

### 12.3.3 Get List of Schedules for a Disk

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

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

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

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<settings>
  <schedule>
    <duration>1</duration>
    <created_at>2011-07-20T15:16:16Z</created_at>
    <target_id>112</target_id>
    <updated_at>2011-07-27T15:16:18Z</updated_at>
    <period>days</period>
    <action>autobackup</action>
    <start_at>2011-07-28T15:16:16Z</start_at>
    <id>33</id>
    <user_id>1</user_id>
    <schedule_logs>
      <schedule_log>
        <created_at>2011-07-27T15:16:18Z</created_at>
        <updated_at>2011-07-27T15:16:18Z</updated_at>
        <schedule_id>33</schedule_id>
        <id>10</id>
        <log_output></log_output>
        <status>complete</status>
      </schedule_log>
      ...
    </schedule_logs>
    <params nil="true"></params>
    <failure_count>0</failure_count>
    <status>enabled</status>
    <target_type>Disk</target_type>
  </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

### 12.3.4 Get List of Virtual Server Schedules

To get a list of schedules for a particular virtual server, use the following methods:

GET /virtual_machines/:id/schedules.xml

GET /virtual_machines/:id/schedules.json

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<schedules 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> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs> 
</schedule> 
</schedule_log> 
</schedule> 
</schedule_logs>
<failure_count type="integer">0</failure_count>
<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>

<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>

<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>
<rotation_period type="integer">1</rotation_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>
  </schedule_log>
</schedule_logs>
<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

12.3.5 Add Schedule to Disk

You can add a schedule to a disk using the following method:

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

XML Request example

curl -i -X POST
http://onapp.test/settings/disks/:disk_id/schedules.xml -d
'\<schedule\><action>autobackup</action><duration>1</duration><period
days</period><rotation_period>1</rotation_period><status>enabled</status><start_at>2014-11-12 10:36</start_at></schedule>' -u
user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example

curl -i -X POST
http://onapp.test/settings/disks/:disk_id/schedules.json -d
'\{"schedule":{"action":"autobackup","duration":"1","period":"days","rotation_period":"1","status":"enabled","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* - number of backups after which the first backup will be deleted
*status* - set enabled to activate a schedule.
*start_at* - set the time when backup scheduling transaction will be created.

### 12.3.5.1 Page history

OnApp 3.5:
- *start_at*

OnApp 3.2:
- *rotation_period*

### 12.3.6 Add Schedule to Virtual Server

To add incremental backup schedule to a virtual server:

```plaintext
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.
12.3.6.1 Page history

OnApp 3.5:
- start_at

OnApp 3.2:
- rotation_period

12.3.7 Edit Disk Schedule

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

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

XML Request example

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -
-d '<schedule><duration>3</duration><period>days</period><rotation_period>1</rotation_period><status>enabled</status><start_at>2014-11-12 10:36</start_at></schedule>' -u user:userpass --url http://onapp.test/settings/schedules/:id.xml

JSON Request example

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -
-d '{"schedule":{"duration":"1","rotation_period":"1","period":"years","status":"enabled","start_at": "2014-11-12 10:36"}}' -u user:userpass --url http://onapp.test/settings/schedules/:id.json

Currently, you can edit the following parameters:
- 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
- status* - set enabled to activate a schedule.
- start_at - set the time when backup scheduling transaction will be created.

12.3.7.1 Page history

OnApp 3.5:
- start_at

OnApp 3.2:
- rotation_period

12.3.8 Edit Virtual Server Schedule

To edit virtual server’s incremental backup schedule, use the following method:
PUT /virtual_machines/:id/schedules/:schedule_id.xml
PUT /virtual_machines/:id/schedules/:schedule_id.json

XML Request example
```
-d '<schedule><duration>6</duration><period>weeks</period><rotation_period>5</rotation_period><enabled>1</enabled><action>autobackup</action><start_at>2014-11-12 10:36</start_at></schedule>'
```

JSON Request example
```
-d '{"schedule":{"duration":"6","period":"weeks","rotation_period":"5","enabled":"1","action":"autobackup","start_at": "2014-11-12 10:36"}}'
```

Currently, you can edit the following parameters:
- **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
- **status** - set enabled to activate a schedule.
- **start_at** - set the time when backup scheduling transaction will be created.

### 12.3.8.1 Page history
OnApp 3.5:
- **start_at**

OnApp 3.2:
- **rotation_period**

### 12.3.9 Delete Disk Schedule
DELETE /settings/schedules/:id.xml
DELETE /settings/schedules/:id.json

XML Request example
```
```

JSON Request example
```
```
http://onapp.test/settings/schedules/:id.json

12.3.10 Delete Virtual Server Schedule

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.
13 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.

13.1 Get List of All VS Backups

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
  <?xml version="1.0" encoding="UTF-8"?>
  <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>lvm</data_store_type>
<id type="integer">1951</id>
<identifier>uml64qyvbzv1kb</identifier>
<img type nil="true"/>
<initiated>days</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 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>

Explanation of the data returned:

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:
**iqn** - volume ISCSI qualified name

### 13.2 Get List of Normal Backups

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**
```
<?xml version="1.0" encoding="UTF-8"?>
<backups type="array">
  <backup>
    <allow_resize_without_reboot
```
<backup>
  <id type="integer">1951</id>
  <identifier>uml64qyvbzv1kb</identifier>
  <image_type nil="true"/>
  <initiated>days</initiated>
  <ignore 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_disk_size>
  <note nil="true"/>
  <operating_system>linux</operating_system>
  <operating_system_distro>ubuntu</operating_system_distro>
  <target_id type="integer">11860</target_id>
  <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>
</backup>
</backups>

**Explanation of the data returned:**

- **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:
ign - volume ISCSI qualified name

13.3 Get List of Incremental Backups

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
   <?xml version="1.0" encoding="UTF-8"?>
   <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 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>
<operating_system_distro>ubuntu</operating_system_distro>
<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>
<brick
</backup>
</backups>

**Explanation of the data returned:**

*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:

13.4 Get List of Disk Backups

To view the list of disk backups:
GET /virtual_machines/:vm_id/disks/:disk_id/backups.xml
GET /virtual_machines/:vm_id/disks/:disk_id/backups.json

XML Request example:
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type:
application/xml' -u user:userpass --url
http://onapp.test/virtual_machines/:vm_id/disks/:disk_id/backups.xml

JSON Request example:
curl -i -X GET -H 'Accept: application/json' -H 'Content-type:
application/json' -u user:userpass --url
http://onapp.test/virtual_machines/:vm_id/disks/:disk_id/backups.json

XML Output example
<?xml version="1.0" encoding="UTF-8"?>
<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>lvm</data_store_type>
<id type="integer">1951</id>
<identifier>uml64qyvbzv1kb</identifier>
<image_type nil="true"/>
<initiated>days</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 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>

Explanation of the data returned:

- **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
**13.5 Create Incremental Backup**

To take an incremental backup, use the following method:

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

**XML Request example**
```bash
```

**JSON Request example**
```bash
```

Where you have to specify the virtual server's ID

**note** - optional backup note

---

**13.6 Create Disk Backup**

To take incremental backups for virtual servers that have incremental backups enabled,
To create a backup of a disk, use the following method:

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

**XML Request example**
```
```

**JSON Request example**
```
```

*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.

## 13.7 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:

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

**XML Request example**
```
```

**JSON Request example**
```
```

Where you have set the ID of a virtual server.
13.8 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.

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 -d '<?xml version="1.0" encoding="UTF-8"?><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/:backup_id/convert.xml

JSON Request example

curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"backup":{"label":"backup_label", "min_disk_size":"20", "min_memory_size":"512"}}' --url http://onapp.test/backups/:backup_id/convert.json

Where you have to specify the following parameters:

- **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".

13.9 Delete Backup

To delete a disk backup:

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

XML Request example


JSON Request example

An HTTP 204 response is returned on success, an HTTP 404 error is returned if a requested backup does not exist.

13.10 Restore Backup

You can restore a disk from a backup, using the following method:

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

XML Request example
```bash
```

JSON Request example
```bash
```

13.11 Add/Edit Backup Note

Use the following API request to update backup with a note:

XML Request example
```bash
```

JSON Request example
```bash
```

Where you have to specify backup ID in the URL.
14 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.

14.1 Get List of Backup Servers

GET /settings/backup_servers.xml
GET /settings/backup_servers.json
Returns the array of backup servers.

XML Output example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<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>
      <backups type="array">
         <backup>
            <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</operating_system>
            <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>
            <backup_size>175264</backup_size>
            <min_disk_size type="integer">5</min_disk_size>
            <identifier>pkg0v4k4n34ym8</identifier>
            <locked type="boolean">false</locked>
         </backup>
      </backups>
   </backup_server>
</backup_servers>
```
Explaination of the data returned:
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

14.2 Get Backup Server Details

This method outputs the details for a particular backup server:
GET /settings/backup_servers/:id.xml
GET /settings/backup_servers/:id.json
XML Output example
<backup_server>
  <label>bk1</label>
  <created_at type="datetime">2012-01-04T1204TI0:18:59+02:0059Z</created_at>
  <backups type="array">
    <backup>
      <marked_for_delete type="boolean">false</marked_for_delete>
      <disk_id type="integer">4097</disk_id>
      <built_at nil="true"/>
      <operating_system_distro>rhel</operating_system_distro>
      <created_at type="datetime">2012-02-11T00:36:17Z</created_at>
      <template_id type="integer">211</template_id>
      <operating_system>linux</operating_system>
      <created_at type="datetime">2012-03-05T13:42:15+02:002-11T00:36:17Z</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="boolean">true</allowed_hot_migrate>
      <backup_size nil="true"/>
      <min_disk_size nil="true"/>
      <identifier>gmkrf5k0s4hsnj</identifier>
      <locked type="boolean">true</locked>
      <built type="boolean">false</built>
    </backup>
    <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">28</backup_server_group_id>
  </backups>
</backup_server>

Explanation of the data returned:
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.

## 14.3 Add Backup Server

To create a backup server where users will be able to store backups and templates, send 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></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"}}'
   -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.

**XML Output example**

```
<?xml version="1.0" encoding="UTF-8"?>
<backup_server>
   <label>az_val_1</label>
</backup_server>
```
14.4 Edit Backup Server

Use the following method to edit a backup server:

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/:id.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></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/:id.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"}}' -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

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.
14.5 Delete Backup Server

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

XML Request example

curl -i -X DELETE -u user:userpass

JSON Request example

curl -i -X DELETE -u user:userpass

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.

14.6 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

You can set the following search parameters:

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.

XML Request example

curl -i -X GET -u 'user:userpass' --url

JSON Request example

curl -i -X GET -u 'user:userpass' --url
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 -i -X GET -u 'user:userpass' --url
```

**JSON Request example**
```
curl -i -X GET -u 'user:userpass' --url
```

**XML Output example**
```xml
<?xml version="1.0" encoding="UTF-8"?>
<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>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>
    <target_type>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>
</backups>
```
<backup_type>normal</backup_type>
<disk_id type="integer">9287</disk_id>
</backup>
</backups>

Explanation of the data returned:
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][hh][mm][ss]Z format
created_at - the date in the [YYYY][MM][DD][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.
14.7 Create Cloud Boot Backup Server

To create a backup server where users will be able to store backups and templates, send 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
  '<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**

```
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"}}' -H 'Accept: application/json' -H 'Content-type: application/json'
```

**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_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
<?xml version="1.0" encoding="UTF-8"?>
<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_server_group_id nil="true"></backup_server_group_id>
  <id type="integer">25</id>
  <enabled type="boolean">true</enabled>
  <backups type="array"/>
  <capacity type="integer">40</capacity>
</backup_server>
```
<ip_address>172.0.0.2</ip_address>
</backup_server>
15 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.

15.1 Get List of Backup Server Zones

To get the list of backup server zones:
GET /settings/backup_server_zones.xml
GET /settings/backup_server_zones.json

XML Response example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<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

15.1.1 Page history

v. 3.1:
- Added the **location_group_id** parameter

15.2 Get Backup Server Zone Details

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

XML Response example  
```xml
<?xml version="1.0" encoding="UTF-8"?>
<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

### 15.2.1 Page history

**v. 3.1:**  
- Added the **location_group_id** parameter

### 15.3 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**  
```bash
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>"
  -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.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

15.3.1 Page history

v. 5.3:
- Added the server_type parameter

v. 3.1:
- Added the location_group_id parameter

15.4 Edit Backup Server Zone

PUT /settings/backup_server_zones/:id.xml
PUT /settings/backup_server_zones/:id.json
XML Request example
```bash
curl -i -X PUT -u user:userpass
  http://onapp.test/settings/backup_server_zones/:id.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'
```

JSON Request example
```bash
curl -i -X PUT -u user:userpass
  http://onapp.test/settings/backup_server_zones/:id.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.

15.4.1 Page history

v. 3.1:
- Added the location_group_id parameter
15.5 Delete Backup Server Zone

To delete a backup server zone, use the following API call:

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

**XML Request example**

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

**JSON Request example**

```bash
curl -i -X DELETE -u user:userpass
http://onapp.test/settings/backup_server_zones/:id.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.

15.6 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

**JSON Request example**

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

**XML Request example**

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

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
```
WHITELIST IPS
- EDIT WHITELISTED IP

15.7 Assign Backup Server to Backup Server Zone

POST
/settings/backup_server_zones/:backup_server_zone_id/backup_servers/:backup_server_id/attach.xml

POST
/settings/backup_server_zones/:backup_server_zone_id/backup_servers/:backup_server_id/attach.json

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

**JSON Request example**

```bash
curl -i -X POST -u user:userpass
```

### 15.8 Unassign Backup Server from Backup Server Zone

**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
```

**JSON Request example**

```bash
curl -i -X POST -u user:userpass
```
16 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.

16.1 Get List of Baremetal Servers

GET /baremetal_servers.xml
GET /baremetal_servers.json

Returns the array of baremetal servers

```xml
<baremetal_servers type="array">
<baremetal_server>
<admin_note nil="true"/>
<allowed_swap type="boolean">true</allowed_swap>
<built type="boolean">true</built>
<cpu_socket nil="true"/>
<cpu_threads nil="true"/>
<cpu_units 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-`
WHITELIST IPS - EDIT WHITELISTED IP

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>

<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">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]T[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

16.2 Get Baremetal Server Details

GET /baremetal_servers/:id.xml
GET /baremetal_servers/:id.json
Returns the array of baremetal servers
  <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"/>
    <cpu_units 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>
<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>
<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">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>
</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
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
16.3 Create Baremetal Server

The management network should be disconnected during the bare metal server deployment.

POST /baremetal_servers.xml
POST /baremetal_servers.json

XML Request example

```xml
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><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_id>63</selected_ip_address_id><required_ip_address_assignment>1</required_ip_address_assignment><recipe_ids type='array'><recipe_id>11</recipe_id></recipe_ids></baremetal_server>'
-u user:password -H 'Accept: application/xml' -H 'Content-Type: application/xml'
```

JSON Request example

```json
curl -i -X POST http://onapp.test/baremetal_servers.json -d
  '{"baremetal_server":{"template_id":"2","label":"test_baremetal","hostname":"test","hypervisor_group_id":"121","hypervisor_id":"38","initial_root_password":"qwaszx","primary_network_group_id":"120","selected_ip_address_id":"63","required_ip_address_assignment":"1","recipe_ids":[]}'}
-u user:password -H 'Accept:application/json' -H 'Content-type: application/json'
```

The following parameters should be sent:

- `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
- `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"
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.

### 16.4 Delete Baremetal Server

DELETE /baremetal_servers/:id.xml
DELETE /baremetal_servers/:id.json

**XML Request example**
curl -i -X DELETE -u user:userpass
    http://onapp.test/baremetal_servers/:id.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request example**
curl -i -X DELETE -u user:userpass
    http://onapp.test/baremetal_servers/:id.json -H 'Accept: application/json' -H 'Content-type: application/json'

Where:

- id - the ID of a baremetal server you want to delete

### 16.5 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

**Add/Edit Admin Note XML Request example**
curl -i -X PUT -u user:userpass
    http://onapp.test/baremetal_servers/:baremetal_server_id.xml -d
    '<baremetal_server><admin_note>agfagwe tiuuytjgh yuytu</admin_note></baremetal_server>' -H 'Accept:application/xml'
    -H 'Content-type:application/xml'

**Add/Edit Admin Note JSON Request example**
curl -i -X PUT -u user:userpass
    http://onapp.test/baremetal_servers/:baremetal_server_id.json -d
    '{"baremetal_server":{"admin_note":"kjfjhjtrtjt"}}' -H
'Accept:application/json' -H 'Content-type:application/json'

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.

Add/Edit User Note XML Request example
   curl -i -X PUT -u user:userpass
          http://onapp.test/baremetal_servers/:baremetal_server_id.xml -d
          '<baremetal_server><note>agfagwe tiiuuytjgh
                      yuytu</note></baremetal_server>' -H 'Accept:application/xml' -H
          'Content-type:application/xml'

Add/Edit User Note JSON Request example
   curl -i -X PUT -u user:userpass
          http://onapp.test/baremetal_servers/:baremetal_server_id.json -d
          '{"baremetal_server":{"note":"kjfjhjtrtjt"}}' -H
          'Accept:application/json' -H 'Content-type:application/json'

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.

16.6 Enable Recovery Mode for Baremetal Server

PUT /baremetal_servers/:id/enable_recovery.xml
PUT /baremetal_servers/:id/enable_recovery.json

XML Request example
   curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password --url
          http://onapp.test/baremetal_servers/:id/enable_recovery.xml

JSON Request example
   curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:password --url
          http://onapp.test/baremetal_servers/:id/enable_recovery.json

Where you have to specify baremetal server's ID.
16.7 Disable Recovery Mode for Baremetal Server

PUT /baremetal_servers/:id/enable_recovery.xml
PUT /baremetal_servers/:id/enable_recovery.json

XML Request example

JSON Request example

Where you have to specify baremetal server's ID.
17 BILLING PLANS

This class manages billing plans, which incorporate prices and resource limits for users. Billing plans can be associated with compute resource, network, data store and backup server zones, as well as template and edge groups. Consequently, these plans enable you to control overall user resource limits, and limits for resources in different zones of the cloud.

Be aware that the maximum price value that you can set is $10^{13}$.

These are the resources you can limit and set prices for, along with the units in which they are measured:

**Limits for Template Store**

<table>
<thead>
<tr>
<th>Limits for Template Store</th>
<th>Pricing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>User VS limits</strong></td>
<td></td>
</tr>
<tr>
<td>Virtual servers (Max)</td>
<td>VS/hour</td>
</tr>
<tr>
<td>Customer networks (Free, Max)</td>
<td>customer network/hour</td>
</tr>
<tr>
<td>Backup (Free, Max)</td>
<td>backup/hour</td>
</tr>
<tr>
<td>Template (Free, Max)</td>
<td>template/hour</td>
</tr>
<tr>
<td>Templates, ISOs &amp; backup storage (Free, Max)</td>
<td>GB/hour</td>
</tr>
<tr>
<td>Autoscaling (Free, Max)</td>
<td>VS/hour</td>
</tr>
<tr>
<td>ISO (Free, Max)</td>
<td>ISO/hour</td>
</tr>
<tr>
<td>Application Server (Max)</td>
<td></td>
</tr>
<tr>
<td>Acceleration (Free, Max)</td>
<td>Accelerated VS/hour</td>
</tr>
<tr>
<td>DRaaS</td>
<td></td>
</tr>
<tr>
<td>Disk Size - GB/hour</td>
<td></td>
</tr>
<tr>
<td>RAM - Mb/hour</td>
<td></td>
</tr>
<tr>
<td>CPU Cores - core/hour</td>
<td></td>
</tr>
<tr>
<td>CPU Shares - percent/hour or CPU Units - unit/hour</td>
<td></td>
</tr>
<tr>
<td>Nodes - unit/hour</td>
<td></td>
</tr>
<tr>
<td>Container Server (Max)</td>
<td></td>
</tr>
<tr>
<td>Template Store(s)</td>
<td></td>
</tr>
</tbody>
</table>
Limits for Recipe Groups

Recipe Group(s)

Limits for Service Add-on Groups

Service Add-on Group(s)

**Limits for Compute Zones**

- CPU cores (Free, Max, On, Off)
- CPU share (Free, Max, On, Off)
- CPU units (Free, Max, On, Off)
- Memory (Free, Max, On, Off)
- CPU Priority (Min, Default)

**Data store zone limits**

- Disk size (Free, Max, On, Off)
- Data read (Free per hour, Price over free units)
- Data written (Free per hour, Price over free units)
- Input requests (Free per hour, Price over free units)
- Output requests (Free per hour, Price over free units)

**Network zone limits**

- IP address (Free, Max, On, Off)
- Port speed (Free, Max, On, Off)
- Data received (free per hour/month, price over free units)
- Data sent (free per hour/month, price over free units)

**Backup server zone limits**

- Backups (Free, Max)

**Pricing**

- CPU core/hour (for VSs on and off)
- CPU share %/hour (for VSs on and off)
- CPU unit/hour (for VSs on and off)
- Mb/hour
- GB/hour
- 1M requests/hour
- GB
- GB
- backup/hour
### Whitelisted IP

- **Backups disk size (Free, Max)**: GB/hour
- **Templates (Free, Max)**: template/hour
- **Templates disk size (Free, Max)**: GB/hour
- **OVAs**: template/hour
- **OVA Disk Size**: GB/hour

### Limits for Edge Groups

- **Edge group**: GB

### Limits for Guaranteed min IOPS

- **Solidfire data store zones (Free, Max, On, Off)**: 1M requests/hour

### Limits for Instance Packages

- **Instance Packages (On, Off)**: VS/hour

For the complete list of billing plan limits along with their details, refer to the [Set Billing Plan Prices And Resource Limits](#) section of the Admin guide.

Starting from the 3.1 release, the master bucket and master template are introduced:

- Master buckets are set for compute zone limits.
- Master templates set for data store zone and network zone limits, respectively.

The master zones are added to each billing plan automatically. When the master bucket or master template limits are applied to a base resource, each virtual server within that base resource (compute resource, data store or network zone) will come under these values.

**NOTE:** You can not edit a base resource which is a part of a master bucket or a master template. The HTTP status 422 will be returned.

The relevant API calls have been added for the master zones management and are added to the Billing Plans section of this guide:

- **Add Base Resource to Master Bucket**
- **Remove Base Resource from Master Bucket**
- **Add Base Resource to Master Template**
- **Remove Base Resource From Master Template**
17.1 Get the List of Billing Plans

To get the list of billing plans created in your cloud, use the following request:
GET /billing/user/plans.xml
GET /billing/user/plans.json

**XML Request example**

**JSON Request example**

**XML Output example**
<?xml version="1.0" encoding="UTF-8"?>
<user_plans type="array">
  <user_plan>
    <id type="integer">2</id>
    <label>Cloud Location manager Billing Plan</label>
    <created_at type="datetime">2015-08-13T15:07:42+03:00</created_at>
    <updated_at type="datetime">2015-08-13T15:07:42+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>
    <type>Billing::User::Plan</type>
    <associated_with_users type="integer">1</associated_with_users>
  </user_plan>
  ...</user_plan>
</user_plans>

Where:
allows_kms - true, if the KMS licensing is allowed for this billing plan, otherwise false
allows_mak - true, if the MAK licensing is allowed, otherwise false
allows_own - true, if adding own licenses is allowed for this billing plan, otherwise false
created_at - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
currency_code - the currency in which the users are charged
id - the billing type ID
label - the billing plan name
monthly_price * - monthly fee for plan usage
show_price - true, if users can see the prices set up for them, otherwise false.
updated_at - the date when the billing plan was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
associated_with_users - the number of users with which this billing plan is associated
type - the type of billing plan

17.1.1 Page history

v5.2:
- Removed the deprecated request method - GET /billing_plans

v4.2:
- Added the alternative request method - GET /billing/user/plans.
- Added the following parameters:
  - type (for method GET /billing/user/plans)

v3.3:
- limit_cpu_units
- limit_free_cpu_units
- use_cpu_units
- price_on_cpu_units
- price_off_cpu_units

v3.1:
- Added the following parameters:
  - associated_with_users
  - is_bucket
  - is_template
  - in_bucket_zone
  - in_template_zone

- Removed the following parameters:
  - is_default
  - use_default
17.2 Get Billing Plan Details

To get the details of billing plan created in your cloud, use one the following request:

GET /billing/user/plans/:id.xml
GET /billing/user/plans/:id.json

XML Request example

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

JSON Request example

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

XML Output example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<user_plan>
  <id type="integer">1</id>
  <label>default billing</label>
  <created_at type="datetime">2015-08-13T15:07:41+03:00</created_at>
  <updated_at type="datetime">2015-08-13T15:07:41+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>
  <type>Billing::User::Plan</type>
  <associated_with_users type="integer">15</associated_with_users>
</user_plan>
```

Where:
- `allows_kms` - true, if the KMS licensing is allowed for this billing plan, otherwise false
- `allows_mak` - true, if the MAK licensing is allowed, otherwise false
- `allows_own` - true, if adding own licenses is allowed for this billing plan, otherwise false
- `created_at` - the date in the [YYYY][MM][DD][hh][mm][ss]Z format
- `currency_code` - the currency in which the users are charged
- `id` - the billing type ID
- `label` - the billing plan name
- `monthly_price` - monthly fee for plan usage
- `show_price` - true, if users can see the prices set up for them, otherwise false.
updated_at - the date when the billing plan was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
associated_with_users - the number of users with which this billing plan is associated
type - the type of billing plan

17.2.1 Page history

v5.2:
- Removed the deprecated request method - GET /billing_plans/:id.

v4.2:
- Added the alternative request method - GET /billing/user/plans/:id.
- Added the following parameters:
  - type

v3.3:
- limit_cpu_units
- limit_free_cpu_units
- use_cpu_units
- price_on_cpu_units
- price_off_cpu_units

v3.1:
Added the following parameters:
- associated_with_users
- is_bucket
- is_template
- in_bucket_zone
- in_template_zone
Removed the following parameters:
- is_default
- use_default
17.3 Add Billing Plan

To add new billing plan, use the following request:
POST /billing/user/plans.xml
POST /billing/user/plans.json

XML Request example
```bash
curl -i -X POST http://onapp.test/billing/user/plans.xml -d'"<user_plan><label>billing_label</label><currency_code>USD</currency_code><monthly_price>10</monthly_price><allows_kms>false</allows_kms><allows_mak>true</allows_mak><allows_own>false</allows_own></user_plan>" -u user:userpass -H 'Accept: application/xml' -H 'Content-type:application/xml'
```

JSON request example
```bash
curl -i -X POST http://onapp.test/billing/user/plans.json -d'{"user_plan":{"label":"billing_label","currency_code":"USD","monthly_price":"10","allows_kms":"false","allows_mak":"true","allows_own":"false"}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type:application/json'
```

Where:
- `label` * - the billing plan name
- `currency_code`* - the currency that users will be charged in within this billing plan (USD by default)
- `monthly_price`* - set monthly fee for plan usage
- `allows_kms` - true, if the KMS licensing is allowed for this billing plan, otherwise false
- `allows_mak` - true, if the MAK licensing is allowed, otherwise false
- `allows_own` - true, if adding own licenses is allowed for this billing plan, otherwise false

Returns 201 on success.

You can't create a billing plan with a code of nonexistent currency.

17.3.1 Page history

v5.2:
- Removed the deprecated request method - POST /billing_plans.

17.4 Edit Billing Plan

To edit an existing plan, use the following request:
PUT /billing/user/plans/:id.xml
PUT /billing/user/plans/:id.json

**XML Request example**
curl -i -X PUT http://onapp.test/billing/user/plans/:id.xml -d
  '<user_plan><label>new_label</label><currency_code>USD</currency_code>
  <monthly_price>10</monthly_price><allows_kms>false</allows_kms>
  <allows_mak>true</allows_mak><allows_own>false</allows_own></user_plan>'
  -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request example**
curl -i -X PUT http://onapp.test/billing/user/plans/:id.json -u
  user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' -d
  '{"user_plan":{"label":"new_label","currency_code":"USD","monthly_price":"10","allows_kms":false,"allows_mak":true,"allows_own":false}}'

Where:
- **label** - the desired billing plan name
- **currency_code** - the code of the currency you're going to charge in.
- **monthly_price** - set monthly fee for plan usage
- **allows_kms** - true, if the KMS licensing is allowed for this billing plan, otherwise false
- **allows_mak** - true, if the MAK licensing is allowed, otherwise false
- **allows_own** - true, if adding own licenses is allowed for this billing plan, otherwise false

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

Editing a billing plan that is associated with more than one user will affect all users attached to it. If you wish to only affect that user then copy the billing plan and associate it only with the single user.

### 17.4.1 Page history

v5.2:
- Removed the deprecated request method - PUT /billing_plans/:id.

### 17.5 Get List of Base Resources.

To view which base resources were added to a particular billing plan, use the following request:
GET /billing/user/plans/:id/resources.xml
GET /billing/user/plans/:id/resources.json
This API call returns only those base resources (and their details), which are assigned to this billing plan. See the Get Base Resources section to learn more about base resources and their details.

**XML Request example**
curl -i -X GET http://onapp.test/billing/user/plans/:id/resources.xml -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request example**
curl -i -X GET http://onapp.test/billing/user/plans/:id/resources.json -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

Returns HTTP status 200 on success

**XML Output example**
```xml
<?xml version="1.0" encoding="UTF-8"?>
<resources type="array">
   <resource>
      <billing_plan_id type="integer">6</billing_plan_id>
      <created_at type="datetime">2013-09-10T13:41:53+00:00</created_at>
      <id type="integer">41</id>
      <master type="boolean">false</master>
      <limit_type>hourly</limit_type>
      <limits>
         <limit_free type="integer">0</limit_free>
         <limit_data_read_free type="integer">0</limit_data_read_free>
         <limit_data_written_free type="integer">0</limit_data_written_free>
         <limit_reads_completed_free type="integer">0</limit_reads_completed_free>
         <limit_writes_completed_free type="integer">0</limit_writes_completed_free>
         <limit nil="true"/>
      </limits>
      <preferences/>
   </resource>
</resources>
```
<?xml version="1.0" encoding="UTF-8"?><resources><resource><billing_plan_id:nil="true"/>6</billing_plan_id><master_resource_id:nil="true"/>40</master_resource_id><created_at:type="datetime">2013-09-10T13:41:53+00:00</created_at><id:type="integer">40</id><master:type="boolean">true</master><limit_type>hourly</limit_type><limits><limit_free_cpu:nil="true"/>0</limit_free_cpu><limit_free_cpu_share:nil="true"/>0</limit_free_cpu_share><limit_free_memory:nil="true"/>0</limit_free_memory><limit_cpu:nil="true"/>0</limit_cpu><limit_cpu_share:nil="true"/>0</limit_cpu_share><limit_memory:nil="true"/>0</limit_memory><limit_default_cpu:nil="true"/>1</limit_default_cpu><limit_default_cpu_share:nil="true"/>100</limit_default_cpu_share><limit_cpu_units:nil="true"/>100</limit_cpu_units><limit_free_cpu_units:nil="true"/>100</limit_free_cpu_units></limits><preferences><use_default_cpu:nil="true"/>false</use_default_cpu><use_default_cpu_share:nil="true"/>false</use_default_cpu_share><use_cpu_units:nil="true"/>false</use_cpu_units></preferences><prices><price_on_cpu>0.00000000</price_on_cpu><price_off_cpu>0.00000000</price_off_cpu><price_on_cpu_share>0.00000000</price_on_cpu_share><price_off_cpu_share>0.00000000</price_off_cpu_share><price_on_memory>0.00000000</price_on_memory><price_off_memory>0.00000000</price_off_memory></prices><target_id:nil="true"/></resource><target_id:nil="true"/></resources>
<resource_name>hypervisor_group</resource_name>
<in_master_zone type="boolean">false</in_master_zone>
</resource>
<resource>
<billing_plan_id type="integer">6</billing_plan_id>
<master_resource_id nil="true"/>
<created_at type="datetime">2013-09-10T13:41:53+00:00</created_at>
$id type="integer">42</id>
</resource>

Where:
resources - an array of billing plan base resources with their details:
- billing_plan_id - the ID of the billing plan
- 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
- id - resource ID
- master - true, if this resource is a master bucket or a template zone, otherwise false.
- label - name of the base resource target
- **resource_name** – the name of the base resource
- **in_master_zone** - true, if this resource belongs to the master bucket or to the master template, otherwise false.
  
  In case of a data store or a network zone that has been added to a master template, the output will show the zone properties that were set before the master template has been applied to the zone. To view the current details of such a zone, you need to view the master template details. For more information refer to Get Master Template/MasterBucket Details.
- **target_id** – ID of a base resource target (e.g. compute zone).
- **prices** – an array of base resource prices
- **limit_type** – hourly or monthly limit type set for the resource
- **limits** - an array of resource limits.
- **limit_free** – the number of GB user gets for free
- **limit_cpu_units** - the total number of CPU units users can get with this plan
- **limit_free_cpu_units** - the number of CPU units get for free
- **use_cpu_units** - true if CPU units are used instead of CPU shares
- **price_on_cpu_units** - price per unit per hour per VSs powered on
- **price_off_cpu_units** - price per CPU unit for VSs powered off

See Get Base Resources for details on each resource limits type.

### 17.5.1 Page history

**v5.2:**
- Removed the deprecated request method - GET /billing_plans/:billing_plan_id/base_resources.

**v4.2:**
- Added the alternative request method - GET /billing/user/plans/:id/resources.
- Introduced new names for some parameters:
  - **resource**
  - **master**
  - **in_master_zone**
  - **master_resource_id**

**v3.3:**
- **limit_cpu_units**
- **limit_free_cpu_units**
- **use_cpu_units**
v3.1:
Added the following parameters:

- associated_with_users
- is_bucket
- is_template
- in_bucket_zone
- in_template_zone

### 17.6 Get Base Resources Details

To get the resource details, use the following request:

GET /billing/user/plans/:id/resources/:id.xml
GET /billing/user/plans/:id/resources/:id.json

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<resource>
  <billing_plan_id type="integer">1194</billing_plan_id>
  <created_at type="datetime">2013-09-11T08:43:17+03:00</created_at>
  <id type="integer">35793</id>
  <master type="boolean">false</master>
  <limit_type>hourly</limit_type>
  <limits>
    <limit_free type="integer">0</limit_free>
  </limits>
  <preferences/>
  <prices>
    <price>0.00000000</price>
  </prices>
  <updated_at type="datetime">2013-09-11T08:43:17+03:00</updated_at>
  <label>Backups</label>
</resource>
```
17.6.1 Where:

resources - an array of billing plan resources with their details:

For all resources

billing_plan_id - the ID of the billing plan

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 - resource ID

limit_type - hourly or monthly limit type set for the resource

label - name of the base resource target

Depending on the resource type, you will also get the following details:

- Backup limits
- Template, ISOs & Backup Storage limits
- Template limits
- Customer networks limits
- Virtual server limits
- Autoscaling limits
- ISO limits
- Acceleration limits
- DRaaS limits
- Application server limits
- Container server limits
- Template Store limits
- Recipe groups limits
- Service Add-on groups limits
- Compute zone limits
- Data store zone limits
- Network zone limits
- Edge groups limits
- Backup server limits
- **MinIOPS resource limits**
- **Instance packages resource limits**

In case of a data store or a network zone that has been added to a master template, the output will show the zone properties that were set before the master template has been applied to the zone. To view the current details of such a zone, you need to view the master template details. For more information refer to Get Master Template/Master BucketDetails.

**Backups limits**

- **limits** an array of resource limits
- **limit_free** the number of backups users can create for free
- **limit** the total amount of backups allowed
- **prices** an array of base resource prices
- **price** price per backup
- **resource_name** the name of the base resource. In this case it is backups

**Template, ISOs & Backup Storage limits**

- **limits** an array of resource limits
- **limit_free** the amount of free disk space (in GB) users can allocate to store backups, ISOs and templates together
- **limit** the total disk space users can allocate to store backups, ISOs and templates together
- **prices** an array of base resource prices
- **price** price per GB
- **resource_name** the name of the base resource. In this case it is storage_disk_size.

**Template limits**

- **limits** an array of resource limits
- **limit_free** the number of custom templates users can create for free
**Limit**

The total amount of custom templates that user can convert from backups.

**Prices**

An array of resource prices.

**Price**

Price per custom template.

**Resource name**

The name of the base resource. In this case it is `template`.

---

**Customer networks limits**

**Limits**

An array of resource limits.

**Limit free**

The number of customer networks users can create for free.

**Limit**

The total amount of customer networks allowed.

**Prices**

An array of resource prices.

**Price**

Price per customer network.

**Resource name**

The name of the base resource. In this case it is `customer_network`.

---

**Virtual server limits**

**Limits**

An array of resource limits.

**Limit**

The total amount of virtual servers allowed.

**Resource name**

The name of the base resource. In this case it is `vm_limit`.

---

**Autoscaling limits**

**Limits**

An array of resource limits.

**Limit free**

The number of VSs using autoscaling a user can create for free.

**Limit**

Maximum number of VS using autoscaling.

**Prices**

An array of resource prices.

**Price**

Price per VS per hour.

**Resource name**

The name of the base resource. In this case it is `- autoscale`.
ISO limits

- **limits**: an array of resource limits
- **limit_free**: the number of ISOs users can upload to the cloud for free
- **limit**: the total amount of ISOs users can upload to the cloud
- **prices**: an array of resource prices
- **price**: price per ISO
- **resource_name**: the name of the base resource. In this case it is ISO

Acceleration limits

- **limits**: an array of resource limits
- **limit_free**: the amount of VSs the user can accelerate for free
- **limit**: the total amount of accelerated VSs allowed
- **prices**: an array of resource prices
- **price**: price per accelerated VS per hour
- **resource_name**: the name of the base resource. In this case it is acceleration

DRaaS limits

- **prices**: an array of additional prices for a VS with DR enabled
- **price_disk_size**: the additional price for disk size per GB per hour
- **price_memory**: the additional price for RAM per MB per hour
- **price_cpus**: the additional price for CPU core per core per hour
- **price_cpu_shares**: the additional price for CPU per percent per hour
- **price_cpu_units**: the additional price for CPU per unit per hour if the compute zone uses CPU units instead of CPU shares
- **price_nodes**: the additional price for node per unit per hour
resource_name

the name of the base resource. In this case it is draas

Application server limits

limits

an array of resource limits

limit

the total amount of application servers allowed

resource_name

the name of the base resource. In this case it is vm_limit

Container server limits

limits

an array of resource limits

limit

the total amount of container servers allowed

resource_name

the name of the base resource. In this case it is Container Server

Template Store limits

target_id

specify the ID of the template group of a preconfigured system template available to users signed up for this billing plan

resource_name

the name of the base resource. In this case it is template_group.

Recipe groups limits

target_id

specify the ID of the recipe group which will be available to users signed up for this billing plan

resource_name

the name of the base resource. In this case it is recipe_group.

Service Add-on groups limits

target_id

specify the ID of the service add-on group which will be available to users signed up for this billing plan

resource_name

the name of the base resource. In this case it is service_addon_group.

Compute zone limits
**limit_type**

hourly or monthly limit type set for the resource

**in_master_zone**

true, if this resource belongs to the master bucket or to the master template, otherwise false.

**master**

true, if this resource is a master bucket or a template zone, otherwise false.

**limits**

an array of limits set up for this resource

**limit_free_cpu**

the limit of CPU users get for free within this billing plan (CPU core/hour)

**limit_free_cpu_share**

the limit of CPU Priority users get for free within this billing plan (in %)

**limit_free_memory**

the amount of free RAM users get for free with this billing plan (Mb/hour)

**limit_cpu**

the total amount of CPU allowed within this billing plan (CPU core/hour)

**limit_cpu_share**

the total of CPU Priority allowed within this billing plan (in %)

**limit_memory**

the entire amount of RAM (Mb/hour)

**limit_default_cpu**

the number of CPU cores users can add to each VS

**limit_default_cpu_share**

CPU priority users can assign to each VS

**limit_cpu_units**

the total number of CPU units users can get with this plan

**limit_free_cpu_units**

the number of CPU units users get for free

**preferences**

the array of settings for all VSs running on compute resources within this zone

**use_default_cpu_share**

true if users cannot see the CPU Share parameter when creating or editing a VS, otherwise false

**use_default_cpu**

true if users cannot see the CPU parameter when creating or editing a VS, otherwise false

**use_cpu_units**

true if CPU units are used instead of CPU shares

**prices**

an array of base resource prices

**price_on_cpu**

price per CPU core when a VS is powered on

**price_off_cpu**

price per CPU core when a VS is powered off
price_on_cpu_share: the price for the resource for powered on VVs
price_off_cpu_share: the price for the resource for powered off VVs
price_on_memory: the price for memory per MB for powered on VS
price_off_memory: the price for memory per MB for powered off VS
price_on_cpu_units: price per unit per hour per VSs powered on
price_off_cpu_units: price per CPU unit for VSs powered off

Data store zone limits
limit_type: hourly or monthly limit type set for the resource
master: true, if this resource is a master bucket or a template zone, otherwise false.
in_master_zone: true, if this resource belongs to the master bucket or to the master template, otherwise false.
limits: an array of limits set up for this resource
limit_free: free disk space on data store zone
limit: maximum available disk space
limit_data_written_free: the amount of data users get for free for write operations (in GB)
limit_data_read_free: the amount of data users get for free for read operations (in GB)
limit_reads_completed_free: the maximum number (in millions) of input requests which can happen at once
limit_writes_completed_free: the maximum number (in millions) of output requests which can happen at once
prices: an array of base resource prices
price_data_read: price per GB of data for read operations
price_data_written: price per GB of data for write operations
price_on: price per GB of disk size, when VS is on
price_off: price per GB of disk size, when VS is off
**price_writes_completed**  
price per million of output requests which can happen at once

**price_reads_completed**  
price per million of input requests which can happen at once

**resource_name**  
data_store_group

**Network zone limits**

**master**  
true, if this resource is a master bucket or a template zone, otherwise false.

**in_master_zone**  
true, if this resource belongs to the master bucket or to the master template, otherwise false.

**limit_type**  
hourly or monthly limit type set for the resource

**limits**  
an array of resource limits

**limit_ip**  
the total amount of IP addresses

**limit_ip_free**  
the amount of IP addresses users get for free

**limit_data_sent_free**  
the amount of data users can send for free

**limit_data_received_free**  
the amount of data users can receive for free

**limit_rate**  
the total available port speed users

**limit_rate_free**  
the port speed users get for free

**prices**  
an array of base resource prices

**price_ip_on**  
price per IP when VS is on

**price_ip_off**  
price per IP when VS is off

**price_rate_on**  
price for port speed (Mbps) when VS is on

**price_rate_off**  
price for port speed (Mbps) when VS is off

**price_data_sent**  
price for sent data per GB per hour

**price_data_received**  
price for received data per GB per hour

**resource_name**  
the name of the base resource. In this case it is network_group.

**Edge groups limits**
**target_id**  
the ID of the edge group that you add to this billing plan

**resource_name**  
the name of the base resource. In this case it is edge_group.

**Backup server limits**

**resource_name**  
the name of the base resource. In this case it is backup_server_group.

**limits**  
an array of resource limits

**limit_backup_free**  
the number of backups user gets for free

**limit_backup**  
the total number of backups allowed

**limit_backup_disk_size_free**  
disk size user gets for free to store their backups

**limit_backup_disk_size**  
maximum backup disk size allowed

**limit_template_disk_size**  
maximum template disk size allowed

**limit_template**  
the total number of templates allowed

**limit_template_free**  
the number of templates user gets for free

**limit_template_disk_size_free**  
template disk size user gets for free

**limit_ova**  
the total number of OVAs allowed

**limit_ova_disk_size**  
maximum OVA disk size allowed

**limit_ova_free**  
the number of OVAs user gets for free

**limit_ova_disk_size_free**  
OVA disk size user gets for free

**prices**  
an array of base resource prices

**price_backup**  
price per backup over limit

**price_template**  
price per template over limit

**price_template_disk_size**  
price per GB of template disk size over limit

**price_ova**  
price per OVA over limit

**price_ova_disk_size**  
price per GB of OVA disk size over limit
MinIOPS resource limits

- **limits**: an array of resource limits
- **limit_free**: the amount of I/O users get for free
- **limit**: the entire amount of I/O operations
- **prices**: an array of resource prices
- **price_on**: the price for unit for powered on data store zone
- **price_off**: the price for unit per powered off data store zone
- **resource_name**: the name of the base resource. In this case it is solid_fire

Instance packages resource limits

- **prices**: an array of prices set for this resource
- **price_on**: the price for unit for powered on data store zone
- **price_off**: the price for unit per powered off data store zone
- **price_overused_bandwidth**: the price for overused bandwidth per GB/hr
- **preferences**: the compute/data store/network zones to which the instance package will apply
- **hypervisor_group_ids**: the IDs of the compute zones limited by the instance package
- **data_store_group_ids**: the IDs of the data store zones limited by the instance package
- **network_group_ids**: the IDs of the network zones limited by the instance package
- **resource_name**: instance_package

17.6.2 Page History

v5.3:

- Added the resource_name and target_id parameters to a new resource group - Service Add-on groups limits
v5.2:
- 'vm_monit' resource name has been changed to 'autoscale'
- Removed the deprecated request method - GET 
  /billing_plans/:billing_plan_id/base_resources/:id.
- Added the following parameters:
  - limit_ova
  - limit_ova_disk_size
  - limit_ova_free
  - limit_ova_disk_size_free
  - price_ova
  - price_ova_disk_size

v5.1:
- Added container server resource
- Added the price_nodes parameter

v4.2:
Added the alternative request method - GET /billing/user/plans/:id/resources.
introduced new names for some parameters
  - resource
  - master
  - in_master_zone
  - master_resource_id

v3.1:
Added the following parameters:
- associated_with_users
- is_bucket
- is_template
- in_bucket_zone
- in_template_zone
17.7 Get Master Template/Master Bucket Details

To get the Master Template/Master Bucket details, use the following method:

GET /billing/user/resources/:id.xml
GET /billing/user/resources/:id.json

You can view the ID of the Master Template/Master Bucket using the following method:

GET /billing/user/plans/:id/resources.xml
GET /billing/user/plans/:id/resources.json

The resource that has the master parameter set to true is a Master Template or a Master Bucket.

There are three types of master resources:
- Master Bucket for compute zones
- Master Template for data store zones
- Master Template for network zones

XML Output example: Master Bucket for Compute Zones

```xml
<?xml version="1.0" encoding="UTF-8"?>
<resource>
  <id type="integer">15407</id>
  <limits>
    <limit_cpu type="integer">105</limit_cpu>
    <limit_cpu_share type="integer">25</limit_cpu_share>
    <limit_cpu_units nil="true"/>
    <limit_memory type="integer">20</limit_memory>
    <limit_free_cpu type="integer">1000</limit_free_cpu>
    <limit_free_cpu_share type="integer">10</limit_free_cpu_share>
    <limit_free_cpu_units type="integer">0</limit_free_cpu_units>
    <limit_default_cpu type="integer">1</limit_default_cpu>
    <limit_min_cpu type="integer">1</limit_min_cpu>
    <limit_min_memory type="integer">128</limit_min_memory>
    <limit_default_cpu_share type="integer">100</limit_default_cpu_share>
    <limit_min_cpu_priority type="integer">1</limit_min_cpu_priority>
  </limits>
  <prices>
    <price_on_cpu type="float">5.0</price_on_cpu>
    <price_off_cpu type="float">2.0</price_off_cpu>
    <price_on_cpu_share type="float">5.0</price_on_cpu_share>
    <price_off_cpu_share type="float">2.0</price_off_cpu_share>
    <price_on_cpu_units type="integer">0</price_on_cpu_units>
    <price_off_cpu_units type="integer">0</price_off_cpu_units>
    <price_on_memory type="float">5.0</price_on_memory>
    <price_off_memory type="float">2.0</price_off_memory>
  </prices>
</resource>
```
17.7.1 Where:

id - resource ID

limits - array of resource limits

limit_cpu - the total amount of CPU allowed within the billing plan (CPU core/hour)

limit_cpu_share - the total of CPU Share allowed within the billing plan (in %)

limit_cpu_units - the total number of CPU units users can get with this plan

limit_memory - the entire amount of RAM (Mb/hour)

limit_free_cpu - the limit of CPU users get for free within the billing plan (CPU core/hour)

limit_free_cpu_share - the limit of CPU Share (%) users get for free within the billing plan (in %)

limit_free_cpu_units - the number of CPU units users get for free

limit_free_memory - the amount of free RAM users get for free with the billing plan (Mb/hour)

limit_default_cpu - the number of CPU cores users can add to each VS

limit_min_cpu - minimum amount of CPU that can be set when creating a VS under the billing plan

limit_min_memory - minimum amount memory that can be set when creating a VS under the billing plan

limit_default_cpu_share - true if users cannot see the CPU Share parameter when creating or editing a VS, otherwise false

limit_min_cpu_priority - minimum amount of CPU priority that can be set when creating a VS under the billing plan
**WHITELIST IPS**

**prices** - array of resource prices:

- **price_on_cpu** - price per CPU core when a VS is powered on
- **price_off_cpu** - price per CPU core when a VS is powered off
- **price_on_cpu_share** - the price for the resource for powered on VSs
- **price_off_cpu_share** - the price for the resource for powered off VSs
- **price_on_cpu_units** - price per unit per hour per VSs powered on
- **price_off_cpu_units** - price per CPU unit for VSs powered off
- **price_on_memory** - the price for memory per MB for powered on VS
- **price_off_memory** - the price for memory per MB for powered off VS

- **target_id** - the ID of the compute zone which will be available to users signed up for the billing plan
- **billing_plan_id** - the ID of the billing plan
- **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

- **limit_type** - hourly or monthly limit type set for the resource. The hourly limit is used for the Master Bucket.

- **preferences** - the array of settings for all VSs running on compute resources within this zone

- **use_default_cpu** - true if users cannot see the CPU parameter when creating or editing a VS, otherwise false

- **use_default_cpu_share** - true if users cannot see the CPU Share parameter when creating or editing a VS, otherwise false

- **use_cpu_units** - true if CPU units are used instead of CPU shares

- **master** - true, if this resource is a master bucket, otherwise false. This parameter applies to compute zone limits only.

- **label** - name of the base resource target, in this case it is 'Master Bucket'

- **resource_name** - the name of the base resource. In this case it is 'hypervisor_group'

- **in_master_zone** - this parameter is not applicable to the Master Bucket resource

**XML Output example: Master Template for Data Store Zones**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<resource>
  <id type="integer">15408</id>
  <limits>
    <limit type="integer">100</limit>
    <limit_free type="integer">10</limit_free>
    <limit_data_read_free type="integer">1000</limit_data_read_free>
  </limits>
</resource>
```
17.7.2 Where:

id - the ID of the billing plan

limits - array of resource limits

limit - maximum available disk space

limit_free - free disk space on data store zone

limit_data_read_free - the amount of data users get for free for read operations (in GB)

limit_data_written_free - the amount of data users get for free for write operations (in GB)

limit_reads_completed_free - the number (in millions) of input requests which can happen at once users get for free

limit_writes_completed_free - the number (in millions) of output requests which can happen at once users get for free

prices - array of resource prices:
OnApp 5.3 API Guide v2

WHITELIST IPS - EDIT WHITELISTED IP

price_on - price per GB of disk size, when VS is on
price_off - price per GB of disk size, when VS is off
price_data_read - price per GB of data for read operations
price_data_written - price per GB of data for write operations
price_reads_completed - price per million of input requests which can happen at once
price_writes_completed - price per million of output requests which can happen at once
target_id - the ID of the data store zone which will be available to users signed up for the billing plan
billing_plan_id - the ID of the billing plan
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
limit_type - hourly or monthly limit type set for the resource. The hourly limit is used for the Master Template.
master - true, if this resource is a master template, otherwise false.
label - name of the base resource target, in this case it is 'Master Template'
resource_name - the name of the base resource. In this case it is 'data_store_group'
in_master_zone - this parameter is not applicable to the Master Template resource

XML Output example: Master Template for Network Zones

```xml
<?xml version="1.0" encoding="UTF-8"?>
<resource>
  <id type="integer">15409</id>
  <limits>
    <limit_ip type="integer">20</limit_ip>
    <limit_rate type="integer">200</limit_rate>
    <limit_rate_free type="integer">111</limit_rate_free>
    <limit_ip_free type="integer">10</limit_ip_free>
    <limit_data_sent_free type="integer">100</limit_data_sent_free>
    <limit_data_received_free type="integer">100</limit_data_received_free>
  </limits>
  <prices>
    <price_rate_on type="float">5.0</price_rate_on>
    <price_rate_off type="float">2.0</price_rate_off>
    <price_ip_on type="float">5.0</price_ip_on>
    <price_ip_off type="float">2.0</price_ip_off>
    <price_data_sent type="float">5.0</price_data_sent>
    <price_data_received type="float">5.0</price_data_received>
  </prices>
  <target_id nil="true"/>
  <billing_plan_id type="integer">704</billing_plan_id>
</resource>
```
<created_at type="datetime">2016-08-02T17:29:50+03:00</created_at>
updated_at type="datetime">2016-08-02T17:35:43+03:00</updated_at>
<limit_type>hourly</limit_type>
<preferences>
</preferences>
<master type="boolean">true</master>
<label>Master Template</label>
<resource_name>network_group</resource_name>
in_master_zone type="boolean">false</in_master_zone>
</resource>

### 17.7.3 Where:

- **id** - resource ID
- **limits** - array of resource limits:
- **limit_ip** - the total amount of IP addresses
- **limit_rate** - the total available port speed
- **limit_rate_free** - the port speed users get for free
- **limit_ip_free** - the amount of IP addresses users get for free
- **limit_data_sent_free** - the amount of data users can send for free
- **limit_data_received_free** - the amount of data users can receive for free
- **prices** - array of resource prices:
  - **price_rate_on** - price for port speed (Mbps) when VS is on
  - **price_rate_off** - price for port speed (Mbps) when VS is off
  - **price_ip_on** - price per IP when VS is on
  - **price_ip_off** - price per IP when VS is off
  - **price_data_sent** - price for sent data per GB per hour
  - **price_data_received** - price for received data per GB per hour
- **target_id** - the ID of the network zone which will be available to users signed up for the billing plan
- **billing_plan_id** - the ID of the billing plan
- **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
- **limit_type** - hourly or monthly limit type set for the resource. The hourly limit is used for the Master Template.
- **master** - true, if this resource is a master template, otherwise false
- **label** - name of the base resource target, in this case it is Master Template
resource_name - the name of the base resource. In this case it is 'network_group'

in_master_zone - this parameter is not applicable to the Master Template resource

17.8 Add Base Resources to Billing Plan.

To add base resources to the billing plan, use the following request:
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

Base resources can be priced differently: some may have different prices, depending whether VS is on or off (Virtual servers base resource limits); some of the resources are charged per unit, regardless if they are on or off (Other base resource limits); another type of resource serves only as a limit to the billing plan, without any charges (Template groups limits, compute zone limits).

17.8.1 Add User VS Limits.

To add user VS limits to a billing plan, use the following request:
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

XML Request example

curl -i -X POST
http://onapp.test/billing/user/plans/:plan_id/resources.xml -d
'<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <resource resource_class="Resource::resource_name"
    limit="30" limit_free="10">0</limit><limit_free><prices><price>10</price></prices></resource>
</resources>'
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example

curl -i -X POST
http://onapp.test/billing/user/plans/:plan_id/resources.json -d
{"resource":{"resource_class":"Resource::resource_name","limits":{"limit":"30","limit_free":"10"},"prices":{"price":"10"}}}
-u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'

Where:
resource_class - the name of the base resource you add to the billing plan in the following format: Resource::resource_name, where the resource_name can be:
resource_name UI Label
**Autoscale**  
Autoscaling limit

**VmLimit**  
Virtual Server limit

**Template**  
Template limit

**StorageDiskSize**  
Templates, ISOs & Backups Storage limit

**Backup**  
Backups limit

**CustomerNetwork**  
Customer network limit

**Templatelso**  
ISO limit

**ApplicationServer**  
Application Server limit

**Acceleration**  
Acceleration limit

**DRaaS**  
DRaaS limits

**ContainerServer**  
Container server limits

For the **VmLimit** resource:

*limit* - sets maximum amount of units of the resource. This parameter affects the number of virtual servers, VSs in Federation and load balancers users can create.

For **DRaaS** base resource:

*price_disk_size* - set the price for disk size per GB per hour

*price_memory* - set the price for RAM per MB per hour

*price_cpus* - set the price for CPU core per core per hour

*price_cpu_shares* - set the price for CPU per percent per hour

*price_cpu_units* - set the price for CPU per unit per hour if the compute zone uses CPU units instead of CPU shares

*price_nodes* - set the price for node per unit per hour

For all other resources:

*limit* - sets maximum amount of units of the resource

*limit_free* - amount of units which are given for free

*price* - price per unit

### 17.8.1.1 Page history

v. 5.2:

- 'VmMonit' resource name has been changed to 'Autoscale'
- Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.

v. 5.1:

- added *price_nodes* parameter
- added **ContainerServer** base resource

v. 4.2:

- added **DRaaS** base resource
- added **ApplicationServer** base resource
- added **Acceleration** base resource
v. 4.0:
- added TemplateIso base resource

v. 3.1:
- added info on CustomerNetwork base resource

17.8.2 Add Limits for Template Store.
To add limits for template store to a billing plan, use the following request:
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

XML Request example
```
curl -i -X POST -u user:userpass -H'Content-type: application/xml'
   -H'Accept: application/xml'
   http://onapp.test/billing/user/plans/:plan_id/resources.xml
   -d'<resource><resource_class>Resource::TemplateGroup</resource_class>
      <billing_plan_id>21</billing_plan_id><target_id>22</target_id><target_type>ImageTemplateGroup</target_type></resource>'
```

JSON Request example
```
curl -i -X POST -u user:userpass -H'Content-type: application/json'
   -H'Accept: application/json'
   http://onapp.test/billing/user/plans/:plan_id/resources.json
   -d'{"resource":{"resource_class":"Resource::TemplateGroup","billing_plan_id":"21","target_id":"22","target_type":"ImageTemplateGroup"}}'
```

Where:
resource_class * - the name of the base resource you add to the billing plan in the following format: Resource::[resource_name], where [resource_name] is TemplateGroup for Limits for Template Store.
billing_plan_id - the ID of the billing plan
target_type* - the type of the group you add to the billing plan limits. For Limits for Template Store, it is ImageTemplateGroup.
target_id* - the ID of the group you add to billing plan limits

Check the ID of the necessary group with the following call:
GET /settings/image_template_groups.xml

17.8.2.1 Page history

v5.2:
Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.
17.8.3 Add Limits for Recipe Groups.

To add limits for recipe groups to a billing plan, use the following request:

POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

XML Request example

curl -i -X POST http://onapp.test/billing/user/plans/:plan_id/resources.xml -d '
  <resource><resource_class>Resource::RecipeGroup</resource_class><target_type>RecipeGroup</target_type><target_id>1</target_id></resource>' -u user:userpass -H 'Accept: application/xml' -H 'Content-type:application/xml'

JSON Request example

curl -i -X POST http://onapp.test/billing/user/plans/:plan_id/resources.json -d '{"resource":{"resource_class":"Resource::RecipeGroup","target_type":"RecipeGroup","target_id":"2"}}' -u user:userpass -H 'Accept: application/json' -H 'Content-type:application/json'

Where:
resource_class * - the name of the base resource you add to the billing plan in the following format: Resource::[resource_name], where [resource_name] is RecipeGroup for Limits for Recipe Group.
billing_plan_id - the ID of the billing plan
target_type* - the type of the group you add to the billing plan limits. For Limits for Recipe Group, it is RecipeGroup.
target_id* - the ID of the group you add to billing plan limits

Check the ID of the necessary group with the following call:
GET /recipe_groups.xml

17.8.3.1 Page history

v5.2:
- Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.

17.8.4 Add Limits for Compute Zones.

When you add a billing plan, the master bucket compute zone is created automatically. You can add your custom compute zones and set the limits and prices for them at any time later.

For details how to add or remove compute zone from the master bucket, use the following
requests:
- Add Base Resource to Master Bucket
- Remove Base Resource from Master Bucket

To limits for compute zones to a billing plan, use the following request:

POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

XML Request example

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

JSON Request example

curl -s -i -X POST
http://onapp.test/billing/user/plans/:plan_id/resources.json -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass


Where:
resource_class * - the name of the base resource you add to the billing plan in the following format: Resource::[resource_name], where [resource_name] is HypervisorGroup for compute zone limits
in_bucket_zone - set 1 to add the compute resource to the master bucket. This parameter applies to compute zone limits only. This parameter is replaced by in_master_zone parameter if you use the first method (POST /billing/user/plans/:plan_id/resources)
use_cpu_units - set 1 to use CPU units instead of CPU shares
WHITELIST IPS

**billing_plan_id** - the ID of the billing plan
**target_type** - the type of the group you add to the billing plan limits, for Limits for compute Zones it is Pack.
**target_id** - the ID of the group (or zone) you add to billing plan limits

Check the ID of the necessary compute zone with the GET /settings/hypervisor_zones call

**limits** - the array of limits for compute zone, all parameters are optional. The empty field sets the unlimited parameter.
- **limit_free_cpu** - set the free amount of CPU
- **limit_free_cpu_share** - set the free amount of CPU shares
- **limit_free_memory** - set the free amount of RAM
- **limit_cpu** - set the CPU limit
- **limit_cpu_share** - set CPU share limit
- **limit_memory** - set the RAM limit
- **limit_cpu_units** - if CPU units are used, set the total number of CPU units users can get with this plan
- **limit_free_cpu_units** - if CPU units are used, set the number of CPU units get for free

**prices** - the array of resource prices. Optional parameters.
- **price_on_cpu** - the price per CPU core/hour when a VS is on
- **price_off_cpu** - the price per CPU core/hour when a VS is off
- **price_on_cpu_share** - the price per CPU core/hour when a VS is on
- **price_off_cpu_share** - the price per CPU core/hour when a VS is off
- **price_on_memory** - the price per Mb/hour for RAM when a VS is on
- **price_off_memory** - the price per Mb/hour for RAM when a VS is off
- **price_on_cpu_units** - if CPU units are used, set price per unit per hour per VSs powered on
- **price_off_cpu_units** - if CPU units are used, set price per CPU unit for VSs powered off

**Page History**

v5.2:
Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.

v.3.3
- **limit_cpu_units**
- **limit_free_cpu_units**
- **use_cpu_units**
- **price_on_cpu_units**
17.8.4.1 Add Compute Zone Base Resource to Master Bucket.

To add a compute zone to the master bucket, use the following request:
```
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json
```

To add or remove existing compute zone base resources from the master bucket, use the following requests:
- Add Base Resource to Master Bucket
- Remove Base Resource from Master Bucket

**XML Request example**
```
curl -i -X POST -u user:userpass -H 'Content-type: application/xml'
    -H 'Accept: application/xml'
    http://onapp.test/billing/user/plans/:plan_id/resources.xml -d
    '<?xml version="1.0" encoding="UTF-8"?>
    <resource>
      <resource_class>Resource::HypervisorGroup</resource_class>
      <target_type>Pack</target_type>
      <in_master_zone>1</in_master_zone>
      <target_id>14</target_id>
    </resource>'
```

**JSON Request example**
```
curl -i -X POST -u user:userpass -H 'Content-type: application/json'
    -H 'Accept: application/json'
    http://onapp.test/billing/user/plans/:plan_id/resources.json -d
    '{"resource":{"resource_class":"Resource::HypervisorGroup",
    "target_type":"Pack", "in_master_zone":"1", "target_id":"14"}}'
```

Where:
- `resource_class`* - the name of the base resource you add to the billing plan in the following format: `Resource::[resource_name]`, where `[resource_name]` is Compute resource Group for compute zone limits
- `target_type`* - the type of the group you add to the billing plan limits, for Limits for compute Zones it is `Pack`
- `in_bucket_zone` - set 1 to add the compute resource to the master bucket. This parameter applies to compute zone limits only. This parameter is replaced by `in_master_zone` parameter if you use the first method (POST /billing/user/plans/:plan_id/resources)
- `target_id`* - the ID of the compute zone which you add to the master bucket
  Check the ID of the required compute zone with the GET /settings/hypervisor_zones call.

**Page history**

v5.2:
- Removed the deprecated request method - POST /billing/plans/:billing_plan_id/base_resources.
17.8.5 Add Limits for Data Store Zones.

By adding data store zone resources to a billing plan, you limit the user only to the data stores in that zone.

When you add a billing plan, the master template is created automatically. You can add your custom data store zones and set the limits and prices for them at any time later.

For details how to add or remove data store zone from the master template, use the following requests:

- Add Base Resource to Master Template
- Remove Base Resource From Master Template

To add data store zones to a billing plan, use the following request:

```
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json
```

**XML Request example**

```bash
curl -i -X POST -u user:userpass -H'Content-type: application/xml'
-H'Accept: application/xml'
```

**JSON Request example**

```bash
curl -i -X POST -u user:userpass -H'Content-type: application/json'
-H'Accept: application/json'
```

Where:
resource_class * - the name of the base resource you add to the billing plan in the following format: Resource::DataStoreGroup
billing_plan_id - the ID of the billing plan
target_type* - type of the group you add to the billing plan limits Pack
target_id * - the ID of the group (or zone) you add to billing plan limits

Check the ID of the necessary data store zone with GET /data_store_zones call.
in_template_zone - true, if this resource belongs to the master template, otherwise false. This parameter applies to data store and network zone limits only. This parameter is replaced by in_master_zone parameter if you use the first method (POST /billing/user/plans/:plan_id/resources)
limits - the array of limits for data store zone, all parameters are optional. The empty field sets the unlimited parameter.
  • limit_free - free disk space on data store zone
  • limit - maximum available disk space
  • limit_data_written_free - the amount of data users get for free for write operations (in GB)
  • limit_data_read_free - the amount of data users get for free for read operations (in GB)
  • limit_reads_completed_free - the maximum number (in millions) of input requests which can happen at once
  • limit_writes_completed_free - the maximum number (in millions) of output requests which can happen at once

prices - the array of resource prices. Optional parameters.
  • price_data_read – price per GB of data for read operations
  • price_data_written - price per GB of data for write operations
  • price_on – price per GB of disk size per hour, when VS is on
  • price_off - price per GB of disk size per hour, when VS is off
  • price_writes_completed – price per million of output requests which can happen at once
  • price_reads_completed - price per million of input requests which can happen at once

limit_type – limit type set for the resource; can be hourly or monthly

17.8.5.1 Page history
v5.2:
Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.

17.8.5.2 Add Data Store Zone Base Resource to Master Template.
To add a data store zone to the master template, use the following request:
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json
To add or remove existing data store zone base resources from the master template, use the following requests:

- **Add Base Resource to Master Template**
- **Remove Base Resource From Master Template**

**XML Request example**
```bash
curl -i -X POST -u user:password -H 'Content-type: application/xml'
   -H 'Accept: application/xml'
   http://onapp.test/billing/user/plans/:plan_id/resources.xml -d
   '<resource><resource_class>Resource::DataStoreGroup</resource_class>
   <target_type>Pack</target_type><in_master_zone>1</in_master_zone>
   <target_id>1</target_id><limit_type>hourly</limit_type></resource>'
```

**JSON Request example**
```bash
curl -i -X POST -u user:password -H 'Content-type: application/json'
   -H 'Accept: application/json'
   http://onapp.test/billing/user/plans/:plan_id/resources.json -d
   '{"resource":{"resource_class":"Resource::DataStoreGroup",
   "target_type":"Pack", "in_master_zone":"1", "target_id":"1",
   "limit_type":"hourly"}}'
```

Where:
- **resource_class** * - the name of the base resource you add to the billing plan in the following format: `Resource::DataStoreGroup`
- **target_type** * - type of the group you add to the billing plan limits `Pack`
- **in_template_zone** - set 1 to add the data store zone to the master template. This parameter applies to data store and network zone limits only. This parameter is replaced by **in_master_zone** parameter if you use the first method (POST /billing/user/plans/:plan_id/resources)
- **target_id** * - the ID of the data store zone which you add to the master template
- **limit_type** - limit type set for the resource; can be hourly or monthly

**Page history**

**v5.2:**
- Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.
17.8.6 Add Limits for Network Zones.

By adding network zone resources to a billing plan, you limit the user only to the network(s) in that zone.

When you add a billing plan, the master template is created automatically. You can add your custom network zones and set the limits and prices for them at any time later.

For details how to add or remove network zone from the master template, use the following requests:

- Add Base Resource to Master Template
- Remove Base Resource From Master Template

To add network zone(s) to a billing plan, use the following request:

POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

**XML Request example**

```bash
curl -i -X POST -u user:userpass -H'Content-type: application/xml'
-H 'Accept: application/xml'
http://onapp.test/billing/user/plans/:plan_id/resources.xml
-d'"resource":{"resource_class":"Resource::NetworkGroup","billing_plan_id":"41","in_master_zone":"1","target_id":"33","target_type":"Pack","limits":{"limit_ip":"20","limit_rate":"20","limit_data_sent_free":"1","limit_rate_free":"2","limit_ip_free":"3","limit_data_received_free":"4"},"prices":{"price_ip_off":"6","price_ip_on":"2","price_rate_off":"3","price_rate_on":"4","price_data_sent":"5","price_data_received":"6"},"limit_type":"hourly"}
```

**JSON Request example**

```bash
curl -i -X POST -u user:userpass -H'Content-type: application/json'
-H 'Accept: application/json'
http://onapp.test/billing/user/plans/:plan_id/resources.json
-d'{"resource":{"resource_class":"Resource::NetworkGroup","billing_plan_id":"41","in_master_zone":"1","target_id":"33","target_type":"Pack","limits":{"limit_ip":"20","limit_rate":"20","limit_data_sent_free":"1","limit_rate_free":"2","limit_ip_free":"3","limit_data_received_free":"4"},"prices":{"price_ip_off":"6","price_ip_on":"2","price_rate_off":"3","price_rate_on":"4","price_data_sent":"5","price_data_received":"6"},"limit_type":"hourly"}}
```

Where:

- `resource_class` - name of the base resource you add to the billing plan in the following format: `Resource::NetworkGroup`
**billing_plan_id** - ID of the billing plan

**target_type** * - type of the group you add to the billing plan limits: Pack

**target_id** * - ID of the network zone you add to billing plan limits

Check the ID of the necessary network zone with GET /settings/network_zones call.

**in_template_zone** true, if this resource belongs to the master template, otherwise false. This parameter applies to data store and network zone limits only. This parameter is replaced by **in_master_zone** parameter if you use the first method (POST /billing/user/plans/:plan_id/resources)

**limits** - the array of limits for network zone, all parameters are optional. The empty field sets the unlimited parameter.

- **limit_ip** - the total amount of IP addresses
- **limit_ip_free** - the amount of IP addresses users get for free
- **limit_data_sent_free** - the amount of data users can send for free
- **limit_data_received_free** - the amount of data users can receive for free
- **limit_rate** - the total available port speed users
- **limit_rate_free** - the port speed users get for free

**prices** - the array of resource prices. Optional parameters.

- **price_ip_on** – price per IP when VS is on
- **price_ip_off** - price per IP when VS is off
- **price_rate_on** – price for port speed (Mbps) when VS is on
- **price_rate_off** - price for port speed (Mbps) when VS is off
- **price_data_sent** – price for sent data per GB
- **price_data_received** – price for received data per GB

**limit_type** – limit type set for the resource; can be hourly or monthly

### 17.8.6.1 Page history

v5.2:
Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.

### 17.8.6.2 Add Network Zone Base Resource to Master Template.

To add a network zone to the master template, use the following request:

POST /billing/user/plans/:plan_id/resources.xml

POST /billing/user/plans/:plan_id/resources.json

To add or remove existing network zone base resources from the master template, use the following requests:

- Add Base Resource to Master Template
- Remove Base Resource From Master Template

**XML Request example**

curl -i -X POST -u user:password -H 'Content-type: application/xml'
WHITELIST IPS

-EDIT

WHITELISTED IP

198

HTTP://ONAPP.TEST/BILLING/USER/PLANS/:PLAN_ID/Resources.xml -d
'<resource><resource_class>Resource::NetworkGroup</resource_class><target_type>Pack</target_type><in_master_zone>1</in_master_zone><target_id>1</target_id><limit_type>hourly</limit_type></resource>'

Where:
resource_class* - name of the base resource you add to the billing plan in the following format: Resource::NetworkGroup

target_type* - type of the group you add to the billing plan limits Pack
in_template_zone - set 1 to add the network store zone to the master template. This parameter is replaced by in_master_zone parameter if you use the first method (POST /billing/user/plans/:plan_id/resources)

target_id* - the ID of the network zone which you add to the master template. This parameter applies to data store and network zone limits only.

Check the ID of the required network zone with the GET /settings/data_store_zones call.

limit_type - limit type set for the resource; can be hourly or monthly

Page history

v5.2:

- Removed the deprecated request method - POST /billing/plans/:billing_plan_id/base_resources.

17.8.7 Add Limits for Edge Groups.

By assigning edge groups to a billing plan, you set the prices for the bandwidth users signed up for this plan consume.

To add edge group(s) to a billing plan, use the following request:

POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

XML request example

curl -i -X POST -d
'<resource><resource_class>Resource::EdgeGroup</resource_class><billing_plan_id>107</billing_plan_id><target_id>1</target_id><target_t
WHITELIST IPS

EDIT WHITELISTED IP

199
**WHITELIST IPS** - **EDIT WHITELISTED IP**

```
mit_template>10</limit_template><limit_template_disk_size_free>2</limit_template_disk_size_free><limit_template_disk_size>10</limit_template_disk_size></limits><prices><price_backup_disk_size>50</price_backup_disk_size><price_template_disk_size>50</price_template_disk_size><price_template>50</price_template></prices></resource>
```

**JSON Request example**

```
curl -i -X POST -u user:userpass http://onapp.test/billing/user/plans/:plan_id/resources.json -d '{"resource":{"resource_class": "Resource::BackupServerGroup","billing_plan_id": "182","target_id": "28","target_type": "Pack","limits": {"limit_backup_free": "2", "limit_backup": "10", "limit_backup_disk_size_free": "2", "limit_backup_disk_size": "10", "limit_template_free": "2", "limit_template": "10", "limit_template_disk_size_free": "2", "limit_template_disk_size": "10"},"prices": {"price_backup": "50","price_backup_disk_size": "50","price_template_disk_size": "50","price_template": "50"}}}' -H 'Accept: application/json' -H 'Content-type: application/json'
```

**Where:**

- **label* - name of the backup server zone assigned to the billing plan**
- **resource_name* - the name of the base resource. In this case it is BackupServerGroup**
- **created at - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format**
- **target_id* - backup server zone id**
- **target_type* - type of the group you add to the billing plan limits: Pack**
- **limits - an array of limits set up for this resource**
- **limit_backup_free - the number of backups user gets for free**
- **limit_backup - the total number of backups allowed**
- **limit_backup_disk_size_free - disk size user gets for free to store their backups**
- **limit_backup_disk_size - maximum backup disk size allowed**
- **limit_template_disk_size - maximum template disk size allowed**
- **limit_template - the total number of templates allowed**
- **limit_template_free - the number of templates user gets for free**
- **limit_template_disk_size_free - template disk size user gets for free**
- **limit_template_disk_size - maximum template disk size allowed**
- **limit_ova - the total number of OVAs allowed**
- **limit_ova_disk_size - maximum OVA disk size allowed**
- **limit_ova_free - the number OVAs user gets for free**
- **limit_ova_disk_size_free - disk size user gets for free to store their OVAs**
- **updated at - the date in the [YYYY][MM][DD]T[hh][mm][ss]Z format**
- **billing_plan_id - the ID of the billing plan**
- **id - resource ID**
- **unit - a unit per which the price is set**
- **price_backup - price per backup over limit**
- **price_template - price per template over limit**
- **price_template_disk_size - price per GB of template disk size over limit**
- **price_backup_disk_size - price per GB of backup disk size over limit**
- **price_ova - price per OVA over limit**
**price_ova_disk_size** - price per GB of OVA disk size over limit

**limit type** - this parameter doesn't mean anything for backup server zone resource

### 17.8.8.1 Page history

**v5.2:**
- Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.
- Added the following parameters:
  - limit_ova
  - limit_ova_disk_size
  - limit_ova_free
  - limit_ova_disk_size_free
  - price_ova
  - price_ova_disk_size

### 17.8.9 Add Limits for minIOPS.

To add limits for minIOPS to a billing plan, use the following request:

POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

**XML Request example**

curl -i -X POST
http://onapp.test/billing/user/plans/:plan_id/resources.xml -d
'"<resource resource_class="Resource::SolidFire"

**JSON Request example**

curl -i -X POST
http://onapp.test/billing/user/plans/:plan_id/resources.json -d
{"resource":{"resource_class":"Resource::SolidFire","billing_plan_id":"4", "target_id":"5", "target_type":"Pack", "limits":{"limit":"10000","limit_free":"600"},"prices":{"price_on": "100", "price_off": "50"}}}" -u user:password -H 'Accept: application/json' -H 'Content-type: application/json'

**Where:**
- resource_class * - the name of the base resource you add to the billing plan in the following format: Resource::SolidFire
- billing_plan_id - the ID of the billing plan
target_type * - the type of the group you add to the billing plan limits: Pack

target_id * - the ID of the data store zone you create a limit for

limits - an array of minimum IO limits:
  • limit_free – the number of IOPS available for customer for free
  • limit – the maximum number of IOPS available

Limits for IOPS are set for a whole data store zone.

prices - an array of resource prices:
  • price_on – price per GB of disk size, when VS is on
  • price_off - price per GB of disk size, when VS is off

17.8.9.1 Page history

v5.2:
Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.

17.8.10 Add Limits for Instance Packages.

To add instance package(s) to a billing plan, use the following request:
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

XML Request example

curl -i -X POST -u user:userpass --url
http://onapp.test/billing/user/plans/:plan_id/resources.xml -d
'<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <resource>
    <resource_class>Billing::Resource::InstanceType</resource_class>
    <target_type>InstanceType</target_type>
    <target_id>10</target_id>
    <hypervisor_group_ids>1</hypervisor_group_ids>
    <data_store_group_ids>7, 37</data_store_group_ids>
    <network_group_ids>11</network_group_ids>
    <price_on>0.3</price_on>
    <price_off>0.2</price_off>
    <price_overused_bandwidth>0.03</price_overused_bandwidth>
  </resource>
</resources>

JSON Request example

curl -i -X POST -u user:userpass --url
http://onapp.test/billing/user/plans/:plan_id/resources.json -d
'{"resource": {"resource_class": "Billing::Resource::InstanceType", "target_type": "InstanceType", "target_id": "10", "hypervisor_group_ids": "1", "data_store_group_ids": "7, 37", "network_group_ids": "11", "price_on": "0.3", "price_off": "0.2", "price_overused_bandwidth": "0.03"}}' -H 'Accept: application/xml' -H 'Content-type: application/xml'

-H 'Accept: application/json' -H 'Content-type: application/json'
Where:
resource_class - name of the base resource you add to the billing plan in the following format: Billing::Resource::InstanceType
target_type - type of the group you add to the billing plan limits: InstanceType
target_id - the ID of the instance package that you add to the billing plan
hypervisor_group_ids - ID(s) of the compute zone(s) to which the instance package will apply
data_store_group_ids - ID(s) of the data store zone(s) to which the instance package will apply
network_group_ids - ID(s) of the network zone(s) to which the instance package will apply
price_on - the hourly price for a VS built using this instance package that is powered on
price_off - the hourly price for a VS built using this instance package that is powered off
price_overused_bandwidth - the price for overused bandwidth per GB

17.8.10.1  Page history
v5.2:
Removed the deprecated request method - POST /billing_plans/:billing_plan_id/base_resources.

17.8.11  Add Limits for Service Add-on Groups
To add limits for service add-on groups to a billing plan, use the following request:
POST /billing/user/plans/:plan_id/resources.xml
POST /billing/user/plans/:plan_id/resources.json

XML Request example
curl -i -X POST
http://onapp.test/billing/user/plans/:plan_id/resources.xml -d
'"resource":{"resource_class":"Resource::ServiceAddonGroup","target_type":"ServiceAddonGroup","target_id":1}" -u user:userpass -H 'Accept: application/xml' -H 'Content-type:application/xml'

JSON Request example
curl -i -X POST
http://onapp.test/billing/user/plans/:plan_id/resources.json -d
'"resource":{"resource_class":"Resource::ServiceAddonGroup", "target_type":"ServiceAddonGroup","target_id":2}" -u user:userpass -H 'Accept: application/json' -H 'Content-type:application/json'

Where:
resource_class * - the name of the base resource you add to the billing plan in the following format: Resource::[resource_name], where [resource_name] is ServiceAddonGroup for Limits for Service Add-on Group.
billing_plan_id - the ID of the billing plan
target_type* - the type of the group you add to the billing plan limits. For Limits for Service Add-on Group, it is ServiceAddonGroup.
target_id* - the ID of the group you add to billing plan limits
- Check the ID of the necessary group with the following call:
  GET /service_addon_groups.xml

17.9 Edit Base Resources

This section provides the info on how to change limits and prices for base resources, together with the examples and parameters explanation.

PLEASE NOTE: You cannot edit a base resource which belongs to the master bucket or master template. HTTP 422 status will be returned.

To change limits and prices for a base resource, use the following request:
PUT /billing/user/resources/:id.xml
PUT /billing/user/resources/:id.json

XML Request example
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.xml -d 
'<resource><limit>5</limit></resource>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.json -d '{"resource": 
{ "limit": "10" }},' -H 'Accept: application/json' -H 'Content-type: application/json'

You can check ID of the resource with GET method.
Returns HTTP 204 response on successful processing, and HTTP 404 when there is no billing plan or base resource with a requested ID, or URL is incorrect.

Where:
resource - an array of billing plan limits and prices depending on the resource type. See below for details:
- Backups limits
- Template, ISOs & Backup Storage limits
- Template limits
- Customer networks limits
- Virtual server limits
- Autoscaling limits
- ISO limits
- **Acceleration limits**
- **DRaaS limits**
- **Application server limits**
- **Container Server limits**
- **Template Store limits**
- **Recipe groups limits**
- **Edge groups limits**
- **instance packages resource limits**

**Backups limits**

- **limit_free** the number of backups users can create for free
- **limit** the maximum number of backups users can create according to their template/backup storage space limit.
- **price** price per backup
- **resource_name** the name of the base resource. In this case it is backups

**Template, ISOs & Backup Storage limits**

- **limit_free** the amount of free disk space (in GB) users can allocate to store backups, ISOs and templates together
- **limit** the total disk space users can allocate to store backups, ISOs and templates together
- **price** price per GB

**Template limits**

- **limit_free** the number of custom templates users can create for free
- **limit** the maximum number of user templates which can be created, according to their template/backup storage space limit.
- **price** price per user template

**Customer networks limits**

- **limit_free** the number of customer networks users can create for free
OnApp 5.3 API Guide    v2

**WHITELIST IPS**

**WHITELISTED IP 206**

- **limit**
  - the total amount of customer networks allowed
- **price**
  - price per customer network

**Virtual server limits**

- **limit**
  - the total amount of virtual servers allowed. This parameter affects the number of virtual servers, VSs in Federation and load balancers users can create.

**Autoscaling limits**

- **limit_free**
  - the number of VSs using autoscaling a user can create for free
- **limit**
  - maximum number of VS using autoscaling
- **price**
  - the price for the VSs using autoscaling (per VS).

**ISO limits**

- **limit_free**
  - the number of ISOs users can upload to the cloud for free
- **limit**
  - the total amount of ISOs users can upload to the cloud, according to their template/ISO/backup storage space limit.
- **price**
  - price per ISO per hour

**Acceleration limits**

- **limit_free**
  - the amount of VSs the user can accelerate for free
- **limit**
  - the total amount of accelerated VSs allowed
- **price**
  - price per accelerated VS per hour

**DRaaS limits**

- **price_disk_size**
  - the additional price for disk size per GB per hour
- **price_memory**
  - the additional price for RAM per MB per hour
- **price_cpus**
  - the additional price for CPU core per core per hour
**price_cpu_shares**  
the additional price for CPU per percent per hour

**price_cpu_units**  
the additional price for CPU per unit per hour if the compute zone uses CPU units instead of CPU shares

**price_nodes**  
the additional price for node per unit per hour

**Application server limits**

**limit**  
the total amount of application servers allowed

**Container Server limits**

**limit**  
the total amount of container servers allowed

**Template Store limits**

**target_id**  
specify the ID of the template group of a preconfigured system template available to users signed up for this billing plan

**Recipe groups limits**

**target_id**  
specify the ID of the recipe group which will be available to users signed up for this billing plan

**Edge groups limits**

**target_id**  
the ID of the edge group that you add to this billing plan

**instance packages resource limits**

**price_on**  
the price for unit for powered on data store zone

**price_off**  
the price for unit per powered off data store zone

**price_overused_bandwidth**  
the price for overused bandwidth per GB

**hypervisor_group_ids**  
the IDs of the compute zones limited by the instance package

**data_store_group_ids**  
the IDs of the data store zones limited by the instance package
network_group_ids  the IDs of the network zones limited by the instance package

- You can update prices and zones for an instance package in billing plan even if this instance package has been used during virtual server creation. instance package price update takes effect on VS's price approximately five minutes after updating.
- If you have VSs created with the instance package on a particular compute zone, you cannot remove this compute zone from instance packages limits.
- If no compute zones are added to limits for instance packages, you can edit limits and add a compute zone, on which instance package VS is already built. Another compute zones can not be added.

17.9.1 Page history

v5.2:
Removed the deprecated request method - PUT /billing_plans/:billing_plan_id/base_resources/:id.

17.10 Edit Limits & Pricing for Compute Zones

To change limits and prices for compute zones, use the following request:
PUT /billing/user/resources/:id.xml
PUT /billing/user/resources/:id.json

For compute zones, you can change the limits for the following resources: CPU limits, CPU share limits, Memory limits, CPU units.

Edit CPU limits XML Request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'"resource":

Edit CPU limits JSON Request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d '{"resource":
{"limit_cpu": "1","limit_free_cpu": "0","price_on_cpu": ""}'
"10","price_off_cpu": "5","limit_default_cpu": "1"}'} -H 'Accept: application/json'-H 'Content-type: application/json'

Where:
limit_cpu - the maximum number of CPU cores they can request under this plan
limit_free_cpu - the number of CPU cores that users get for free
price_on_cpu - the price per CPU core per hour, for VSs powered on
price_off_cpu - the price per CPU core per hour for VSs powered off
limit_default_cpu - the default values for CPU cores for each VS, which set the amount of CPU cores automatically added per each VS

Use the following request, to reset CPU limits to default:
XML Request example:
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.xml -d '<resource><use_default_cpu>1</use_default_cpu></resource>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

Json request example:
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.json -d '{"resource": {"use_default_cpu": "true"}}' -H 'Accept: application/json' -H 'Content-type: application/json'

Edit CPU Share limits XML Request example
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.xml -d '<resource><limit_cpu_share>1</limit_cpu_share><limit_free_cpu_share>0</limit_free_cpu_share><price_on_cpu_share>10</price_on_cpu_share><price_off_cpu_share>5</price_off_cpu_share><limit_default_cpu_share>100</limit_default_cpu_share></resource>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

Edit CPU Share limits JSON Request example
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.json -d '{"resource": {"limit_cpu_share": "10","limit_free_cpu_share": "1","price_on_cpu_share": "100","price_off_cpu_share": "50","limit_default_cpu_share": "100"}}' -H 'Accept: application/json'

Where:
limit_cpu_share - the maximum CPU priority % the users can request under this plan
limit_free_cpu_share - the CPU priority % that users get for free

price_on_cpu_share - the prices per CPU priority % per hour, for VSs powered on
**price_off_cpu_share** - the prices per CPU priority % per hour, for VSs powered off

**limit_default_cpu_share** - the default values for CPU share for each VS, which set the amount of CPU share automatically assigned to each VS.

Use the following request, to reset CPU Share limits to default:

**XML Request example:**
```bash
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'<?xml version="1.0" encoding="UTF-8"?>
<resource><use_default_cpu_share>1</use_default_cpu_share></resource>'
-H 'Accept: application/xml' -H 'Content-type:
application/xml'
```

**Json request example:**
```bash
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
'{"resource":
{"use_default_cpu_share": "true"}}' -H 'Accept: application/json' -H
'Content-type: application/json'
```

**Edit Memory limits XML Request example**
```bash
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'<?xml version="1.0" encoding="UTF-8"?>
<resource><limit_memory>128</limit_memory><limit_free_memory>0</limit_free_memory><price_on_memory>10</price_on_memory><price_off_memory>5</price_off_memory></resource>'
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

**Edit Memory limits JSON Request example**
```bash
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
'{"resource":
{"limit_memory": "256","limit_free_memory": "10","price_on_memory": "100","price_off_memory": "50"}}' -H 'Accept:application/json' -H 'Content-type: application/json'
```

**Where:**
- **limit_memory** - the maximum RAM users can request under this plan
- **limit_free_memory** - the amount of RAM users get for free
- **price_on_memory** - the prices for RAM for VSs powered on
- **price_off_memory** - the prices for RAM for VSs powered off. RAM is measured in MB and priced per hour.

You can check ID of the Compute Zone resource with GET method.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no billing plan or base resource with a requested ID, or URL is incorrect.
**Edit CPU Units limits XML Request example**
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.xml -d
  '<resource><limit_cpu_units>1000</limit_cpu_units><limit_free_cpu_units>0</limit_free_cpu_units><price_on_cpu_units>10</price_on_cpu_units><price_off_cpu_units>5</price_off_cpu_units></resource>'

Where:
- **limit_cpu_units** - if CPU units are used, set the total number of CPU units users can get with this plan
- **limit_free_cpu_units** - if CPU units are used, set the number of CPU units get for free
- **price_on_cpu_units** - if CPU units are used, set price per unit per hour per VSs powered on
- **price_off_cpu_units** - if CPU units are used, set price per CPU unit for VSs powered off

**Page History**
v5.2:
Removed the deprecated request method - PUT /billing_plans/:billing_plan_id/base_resources/:id.

v.3.3
- **limit_cpu_units**
- **limit_free_cpu_units**
- **use_cpu_units**
- **price_on_cpu_units**
- **price_off_cpu_units**
17.11 Edit Limits & Pricing for Data Store Zones

To change limits and prices for Data store zones, use the following request:

PUT /billing/user/resources/:id.xml
PUT /billing/user/resources/:id.json

You can change the following limits and pricing for data store zone limits: Disk Size, Data read, Data written, Input requests, Output requests.

Edit Disk size limits XML request example

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
  '<resource><limit>60</limit><limit_free>10</limit_free><price_on>100</price_on><price_off>50</price_off></resource>'
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

Edit Disk size limits JSON request example

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
  '{"resource":"limit": "6","limit_free": "0","price_on": "10","price_off": "5"}'
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:
- limit - the maximum disk space users can request under this plan
- limit_free - the disk space that users get for free
- price_on - the price per GB of disk size per hour, when VS is on
- price_off - the price per GB of disk size per hour, when VS is off

Edit Data read limits XML request example

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
  '<resource><limit_data_read_free>20</limit_data_read_free><price_data_read>100.00000000</price_data_read></resource>'
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

Edit Data read limits JSON request example

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
  '{"resource":"limit_data_read_free": "2","price_data_read": "10.00000000"}'
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:
- limit_data_read_free - the amount of data read users can send for free
- price_data_read - the price over free units

Edit Data written limits XML request example

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
  '<resource><limit_data_written_free>50</limit_data_written_free><price_data_written>50.00000000</price_data_written></resource>'
-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

Edit Data written limits JSON request example

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
  '{"resource":"limit_data_written_free": "5","price_data_written": "50.00000000"}'
-H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:
- limit_data_written_free - the amount of data written users can send for free
- price_data_written - the price over free units
WHITELIST IPS

Edit Data written limits JSON request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'"resource": {"limit_data_written_free": 20,"price_data_written":
"100.00000000"}'} -H 'Accept: application/json' -H 'Content-type: application/json'

Where:
limit_data_written_free - the amount of data written users can send for free
price_data_written - the price over free units

Edit Input requests limits XML request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'"resource": {"limit_reads_completed_free": 20,"price_reads_completed":
"100.00000000"}'} -H 'Accept: application/json' -H 'Content-type: application/json'

Where:
limit_reads_completed_free - the number of input requests per hour users can get for free.
price_reads_completed - the price over free units. Input requests are measured in millions and priced per million requests.

Edit Output requests limits XML request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'"resource": {"limit_writes_completed_free":
20,"price_writes_completed": "100.00000000"}'} -H 'Accept: application/xml' -H 'Content-type: application/xml'

Edit Output requests limits JSON request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
'"resource": {"limit_writes_completed_free":
20,"price_writes_completed": "100.00000000"}'} -H 'Accept: application/json' -H 'Content-type: application/json'
Where:

- `limit_writes_completed_free` - the number of output requests per hour users can get for free.
- `price_writes_completed` - the price over free units. Output requests are measured in millions and priced per million requests.

You can check ID of the Data store zone resource with GET method.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no billing plan or base resource with a requested ID, or URL is incorrect.

17.11.1 Page history

v5.2:
Removed the deprecated request method - PUT /billing_plans/:billing_plan_id/base_resources/:id.

17.12 Edit Limits & Pricing for Network Zones

To change limits and prices for Network zones, use the following request:

PUT /billing/user/resources/:id.xml
PUT /billing/user/resources/:id.json

For Network zones, you can change the limits for the following resources: IP Address, Port Speed, Data received, Data sent.

**Edit IP Address limits XML request example**
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.xml -d
'&lt;resource&gt;&lt;limit_ip&gt;20&lt;/limit_ip&gt;&lt;limit_ip_free&gt;10&lt;/limit_ip_free&gt;&lt;price_ip_on&gt;100&lt;/price_ip_on&gt;&lt;price_ip_off&gt;50&lt;/price_ip_off&gt;&lt;/resource&gt;' -H 'Accept: application/xml' -H 'Content-type: application/xml'

**Edit IP Address limits JSON request example**
curl -i -X PUT -u user:userpass --url http://onapp.test/billing/user/resources/:id.json -d
'{"resource": {"limit_ip": "2","limit_ip_free": "1","price_ip_on": "10","price_ip_off": "5"}}' -H 'Accept: application/json' -H 'Content-type: application/json'
Where:

\textit{limit\_ip} - the maximum number of IP addresses users can request under this plan
\textit{limit\_ip\_free} - the number of IP addresses users get for free
\textit{price\_ip\_on} - the price per Mbps per hour for VSs powered on
\textit{price\_ip\_off} - the price per Mbps per hour for VSs powered off

\textbf{Edit Port speed limits XML request example}

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'<resource><limit_rate>99</limit_rate><limit_rate_free>50</limit_rate_free><price_rate_on>100</price_rate_on><price_rate_off>50</price_rate_off></resource>' -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

\textbf{Edit Port speed limits JSON request example}

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
'{"resource": {"limit_rate": "99","limit_rate_free": "50","price_rate_on": "10","price_rate_off": "5"}}' -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

\textit{limit\_rate} - the maximum port speed amount user can request
\textit{limit\_rate\_free} - the amount of port speed user gets for free
\textit{price\_rate\_on} - the price per Mb per hour for VSs powered on
\textit{price\_rate\_off} - the price per Mb per hour for VSs powered off

\textbf{Edit Data received limits XML request example}

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'<resource><limit_data_received_free>10</limit_data_received_free><price_data_received>100.00000000</price_data_received></resource>' -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

\textbf{Edit Data received limits JSON request example}

```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
'{"resource": {"limit_data_received_free": "1","price_data_received": "100.00000000"}}' -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:

\textit{limit\_data\_received\_free} - the amount of data received users can send for free
\textit{price\_data\_received} - the price over free units. Data received is priced per GB.
Edit Data sent limits XML request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
'<resource><limit_data_sent_free>100</limit_data_sent_free><price_data_sent>100.00000000</price_data_sent></resource>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

Edit Data sent limits JSON request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
'{"resource": {"limit_data_sent_free": "10","price_data_sent": "100.00000000"}}' -H 'Accept: application/json' -H 'Content-type: application/json'

Where:

*limit_data_sent_free* - the amount of data sent users can send for free
*price_data_sent* - the price over free units. Data sent is measured in GB and priced per GB.

You can check ID of the Network zone resource with GET method.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no billing plan or base resource with a requested ID, or URL is incorrect.

17.12.1 Page history

v5.2:
Removed the deprecated request method - PUT /billing_plans/:billing_plan_id/base_resources/:id.

17.13 Edit Limits & Pricing for Backup Server Zones

To change limits and prices for Backup Server zones, use the following request:

PUT /billing/user/resources/:id.xml
PUT /billing/user/resources/:id.json

For Backup server zones, you can change the limits for the following resources: Backups, Backup disk size, Templates, Template disk size, OVAs, OVA disk size.

Edit Backups limits XML request example
WHITELIST IPS

EDIT

WHITELISTED IP

217

OnApp 5.3 API Guide    v2

WHITELIST IPS - EDIT WHITELISTED IP  217
Edit Templates limits JSON request example
```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
  '{"resource": {"limit_template": "100","limit_template_free":
    "10","price_template": "100.00000000"}}' -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:
- **limit_template** - the total amount of templates users can have
- **limit_template_free** - the amount of templates users can send for free
- **price_template** - the price for each template over the free limit

Edit Templates disk size limits XML request example
```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
```

Where:
- **limit_template_disk_size** - the total amount of template disk size users can receive for free
- **limit_template_disk_size_free** - the amount of template disk size users can receive for free
- **price_template_disk_size** - the prices for each GB over the limit

Edit OVAs limits XML request example
```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
```

Edit Templates limits JSON request example
```
curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
  '{"resource": {"limit_ova": "100","limit_ova_free":
    "10","price_ova": "100.00000000"}}' -H 'Accept: application/json' -H 'Content-type: application/json'
```

Where:
- **limit_ova** - the total amount of OVAs users can have
- **limit_ova_free** - the amount of OVAs users can send for free
- **price_ova** - the price for each OVA over the free limit
H 'Content-type: application/json'

Where:
limit_ova – the total number of OVAs allowed
limit_ova_free – the number OVAs user gets for free
price_ova – price per OVA over limit

Edit OVA disk size limits XML request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.xml -d
  '<resource><limit_ova_disk_size>10</limit_ova_disk_size>
  <limit_ova_disk_size_free>5</limit_ova_disk_size_free>
  <price_ova_disk_size>10.0000000</price_ova_disk_size></resource>' -H 'Accept:
application/xml' -H 'Content-type: application/xml'

Edit Templates disk size limits JSON request example

curl -i -X PUT -u user:userpass --url
http://onapp.test/billing/user/resources/:id.json -d
  '{"resource": {"limit_ova_disk_size": "100",
  "limit_ova_disk_size_free": "50",
  "price_ova_disk_size": "100.0000000"}}' -H 'Accept: application/json' -H 'Content-type: application/json'

Where:
limit_ova_disk_size – maximum OVA disk size allowed
limit_ova_disk_size_free - disk size user gets for free to store their OVAs
price_ova_disk_size - price per GB of OVA disk size over limit

You can check ID of the Backup Server zone resource with GET method.

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no billing plan or base resource with a requested ID, or URL is incorrect.

17.13.1 Page history

v5.2:

• Removed the deprecated request method - PUT
  /billing_plans/:billing_plan_id/base_resources/:id.

• Added the following parameters:
  
  o limit_ova
  o limit_ova_disk_size
  o limit_ova_free
  o limit_ova_disk_size_free
  o price_ova
  o price_ova_disk_size
17.14 Delete Base Resource From Billing Plan

To delete a base resource from billing plan, use the following request:

DELETE /billing/user/resources/:id.xml
DELETE /billing/user/resources/:id.json

**XML Request example**

curl -i -X DELETE -u user:userpass
http://onapp.test/billing/user/resources/:id.xml

**JSON Request example**

curl -i -X DELETE -u user:userpass
http://onapp.test/billing/user/resources/:id.json

Where you have to specify ID of a billing plan and ID of a resource you want to delete.

To delete a CDN edge group which is associated with CDN resources, add the force=1 parameter to the request, as shown in the examples below.

**XML Request example**

curl -i -X DELETE -u user:userpass
http://onapp.test/billing/user/resources/:id.json -H
'Accept:application/xml' -d '<force>1</force>' -H 'Content-type:application/xml'

**JSON Request example**

curl -i -X DELETE -u user:userpass
http://onapp.test/billing/user/resources/:id.json -d
'{"force":"1"}' -H 'Accept:application/json' -H 'Content-type:application/json'

Returns HTTP 204 response on successful processing, and HTTP 404 when there is no billing plan or base resource with a requested ID, or URL is incorrect.

**Page History**

v5.2:
- Removed the deprecated request method - DELETE /billing_plans/:billing_plan_id/base_resources/:id.

v. 3.1:
- info on force parameter, that allows to delete edge group that is associated with CDN resources.
17.15 Add Base Resource to Master Bucket

To add base resource to the master bucket, use the following request:

**PUT /billing/user/resources/:id.xml**
**PUT /billing/user/resources/:id.json**

Currently the master bucket is used for compute zone base resource limits only.

**XML Request example**

```bash
curl -i -X PUT -H 'Content-Type: application/xml' -H 'Accept: application/xml' -u user:userpass -d '<resource><in_master_zone>1</in_master_zone></resource>'
http://onapp.test/billing/user/resources/:id.xml
```

**JSON Request example**

```bash
curl -i -X PUT -H 'Content-Type: application/json' -H 'Accept: application/json' -u user:userpass -d '{"resource":{"in_master_zone":"1"}}'
http://onapp.test/billing/user/resources/:id.json
```

Where:

- **in_master_zone** - set 1 to add this resource to the master template.

17.15.1 Page history

**v5.2:**
- Removed the deprecated request method - PUT /billing_plans/:billing_plan_id/base_resources/:id.

17.16 Remove Base Resource from Master Bucket

To remove base resource from the master bucket, use the following request:

**PUT /billing/user/resources/:id.xml**
**PUT /billing/user/resources/:id.json**

Currently the master bucket is used for compute zone base resource limits only.
XML Request example

curl -i -X PUT -H 'Content-Type: application/xml' -H 'Accept: application/xml' -u user:userpass -d '<resource><in_master_zone>0</in_master_zone></resource>'
http://onapp.test/billing/user/resources/:id.xml

JSON Request example

curl -i -X PUT -H 'Content-Type: application/json' -H 'Accept: application/json' -u user:userpass -d '{"resource":{"in_master_zone":"0"}}'
http://onapp.test/billing/user/resources/:id.json

Where:
in_master_zone - leave this field empty to remove this resource from the master bucket.

17.16.1 Page history

v5.2:
- Removed the deprecated request method - PUT /billing_plans/:billing_plan_id/base_resources/:id.

17.17 Add Base Resource to Master Template

To add base resource to the master template, use the following request:

PUT /billing/user/resources/:id.xml
PUT /billing/user/resources/:id.json

Currently the master template is used for data store zone and network zone base resource limits only.

XML Request example

curl -i -X PUT -H 'Content-Type: application/xml' -H 'Accept: application/xml' -u user:userpass -d '<resource><master_resource_id>25</master_resource_id></resource>'
http://onapp.test/billing/user/resources/:id.xml

**JSON Request example**
```
curl -i -X PUT -H 'Content-Type: application/json' -H 'Accept: application/json' -u user:userpass -d '{"resource":{"master_resource_id":"25"}}'
```
http://onapp.test/billing/user/resources/:id.json

Where:
*master_resource_id* - ID of the master template.

### 17.17.1 Page history

**v5.2:**
- Removed the deprecated request method - PUT /billing_plans/:billing_plan_id/base_resources/:id.

### 17.18 Remove Base Resource From Master Template

To remove base resource from the master template, use the following request:

**PUT /billing/user/resources/:id.xml**
**PUT /billing/user/resources/:id.json**

Currently the master template is used for data store zone and network zone base resource limits only.

**XML Request example**
```
curl -i -X PUT -H 'Content-Type: application/xml' -H 'Accept: application/xml' -u user:userpass -d '<resource><in_master_zone>0</in_master_zone></resource>'
```
http://onapp.test/billing/user/resources/:id.xml

**JSON Request example**
```
curl -i -X PUT -H 'Content-Type: application/json' -H 'Accept: application/json' -u user:userpass -d '{"resource":{"in_master_zone":"0"}}'
```
http://onapp.test/billing/user/resources/:id.json
Where:
in_master_zone - set 0 to remove this resource from the master template.

17.18.1 Page history

v5.2:
- Removed the deprecated request method - PUT
  /billing_plans/:billing_plan_id/base_resources/:id.

17.19 Delete Billing Plan

To delete a billing plan, use the following request:
DELETE /billing/user/plans/:plan_id.xml
DELETE /billing/user/plans/:plan_id.json

XML Request example
curl -i -X DELETE http://onapp.test/billing/user/plans/:plan_id.xml
-u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example
curl -i -X DELETE
http://onapp.test/billing/user/plans/:plan_id.json -u user:userpass
-H 'Accept: application/json' -H 'Content-Type: application/json'

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

Editing a billing plan that is associated with more than one user will affect all users attached
to it. If you wish to only affect that user then copy the billing plan and associate it only with
the single user.

17.19.1 Page history

v5.2:
- Removed the deprecated request method - DELETE /billing_plans/:id.
18  BLUEPRINTS

Blueprints allow to create and manage multiple VMware virtual servers using imported VMware vApps images (blueprint templates). Blueprint is a set of VMware virtual servers managed as a single multi-tiered application.

18.1  Get List of Blueprints

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

GET /blueprints.xml
GET /blueprints.json

XML Request example

```
```

JSON Request example

```
```

XML Output example

```
<?xml version="1.0" encoding="UTF-8"?>
<blueprints type="array">
  <blueprint>
    <blueprint_template_id type="integer">1</blueprint_template_id>
    <created_at type="datetime">2013-06-04T15:31:25+03:00</created_at>
    <customer_network_id type="integer">14</customer_network_id>
    <data_store_id type="integer">5</data_store_id>
    <id type="integer">9</id>
    <identifier>x12wptaopx61sh</identifier>
    <label>test</label>
    <state>off</state>
    <updated_at type="datetime">2013-06-04T15:31:25+03:00</updated_at>
  </blueprint>
</blueprints>
```
<user_id type="integer">1</user_id>
</blueprint>
</blueprints>

Where:
- `blueprint_template_id` - ID of a template the blueprint is built from
- `created_at` - the date when the blueprint was created in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- `updated_at` - the date when the blueprint was updated in the [YYYY][MM][DD][T][hh][mm][ss]Z format
- `customer_network_id` - ID of a customer network
- `data_store_id` - ID of a data store the blueprint belongs to
- `id` - blueprint ID
- `identifier` - blueprint identifier
- `label` - blueprint name
- `state` - blueprint state: on or off
- `user_id` - ID of a user the blueprint belongs to

18.2 Get Blueprint Details

To get the list of all blueprints, use the following request:
GET /blueprints/:blueprint_id.xml
GET /blueprints/:blueprint_id.json

XML Request example
```
```

JSON Request example
```
```

XML Output example
```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint>
  <blueprint_template_id type="integer">1</blueprint_template_id>
  <created_at type="datetime">2013-06-04T15:31:25+03:00</created_at>
  <customer_network_id type="integer">14</customer_network_id>
  <data_store_id type="integer">5</data_store_id>
  <id type="integer">9</id>
  <identifier>x12wptaopx6lsh</identifier>
</blueprint>
<label>test</label>
<label>off</label>

<updated_at type="datetime">2013-06-04T15:31:25+03:00</updated_at>

Where:

- **blueprint_template_id** - ID of a template the blueprint is built from
- **created_at** - the date when the blueprint was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- **updated_at** - the date when the blueprint was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **customer_network_id** - ID of a customer network
- **data_store_id** - ID of a data store the blueprint belongs to
- **id** - blueprint ID
- **identifier** - blueprint identifier
- **label** - blueprint name
- **state** - blueprint state: on or off
- **user_id** - ID of a user the blueprint belongs to

### 18.3 Add Blueprint

Use the following request to add new blueprint:

**POST /blueprints.xml**

**POST /blueprints.json**

**XML Request example**

curl -i -u user:userpass -X POST http://onapp.test/blueprints.xml -d '

`<blueprint>
  <label>johnblueprint</label>
  <blueprint_template_id>14</blueprint_template_id>
  <data_store_id>5</data_store_id>
  <customer_network_id>116</customer_network_id>
  <required_startup>1</required_startup>
</blueprint>`

- **Accept: application/xml** -H 'Content-type: application/xml'

**JSON Request example**


Where:

- **label** - blueprint name
- **blueprint_template_id** - ID of the template to build the blueprint from
**data_store_id** - ID of a data store to which the blueprint will be assigned
**customer_network** - ID of the customer network which will be used for this blueprint
**required_startup** - set 1 to start up the blueprint after creation

**XML Output example**
```xml
<?xml version="1.0" encoding="UTF-8"?>
<blueprint>
  <blueprint_template_id type="integer">14</blueprint_template_id>
  <built type="boolean">false</built>
  <created_at type="datetime">2013-08-28T16:13:55+03:00</created_at>
  <customer_network_id type="integer">116</customer_network_id>
  <data_store_id type="integer">5</data_store_id>
  <hypervisor_id nil="true"/>
  <id type="integer">59</id>
  <identifier>tdo20mf3ue792s</identifier>
  <label>blueprint</label>
  <state>off</state>
  <updated_at type="datetime">2013-08-28T16:13:55+03:00</updated_at>
  <user_id type="integer">618</user_id>
</blueprint>
```

### 18.4 Delete Blueprint

Use the following request to add new blueprint:

DELETE /blueprints/blueprint_id.xml
DELETE /blueprints/blueprint_id.json

**XML Request example**
```
```

**JSON Request example**
```
```

Where you need to specify ID of a blueprint you want to delete.
19  BLUEPRINT TEMPLATE GROUPS

Blueprint template groups allow OnApp administrators to organize individual templates into groups for convenient template management. Blueprint templates are not associated with billing plans.

19.1 Get List of Blueprint Template Groups

To get the list of blueprint template groups, use the following request:

GET /blueprint_template_groups.xml
GET /blueprint_template_groups.json

XML Request example:

JSON Request example:

XML Output example

<?xml version="1.0" encoding="UTF-8"?>
<objects type="array">
<object>
  <id type="integer">6</id>
  <label>in_blueprints</label>
  <parent_id nil="true"/>
  <lft type="integer">1</lft>
  <rgt type="integer">2</rgt>
  <depth type="integer">0</depth>
  <created_at type="datetime">2013-06-05T14:42:37+00:00</created_at>
  <updated_at type="datetime">2013-06-05T14:42:37+00:00</updated_at>
  <children type="array"/>
</object>
</objects>
<relations type="array">
  <relation>
    <id type="integer">8</id>
    <blueprint_template_id type="integer">12</blueprint_template_id>
    <blueprint_template_group_id type="integer">6</blueprint_template_group_id>
    <created_at type="datetime">2013-06-05T14:42:43+00:00</created_at>
    <updated_at type="datetime">2013-06-05T14:42:43+00:00</updated_at>
    <blueprint_template>
      <id type="integer">12</id>
      <label>vmware_template_vapp</label>
      <user_id type="integer">1</user_id>
      <created_at type="datetime">2013-06-05T14:41:13+00:00</created_at>
      <updated_at type="datetime">2013-06-05T14:41:13+00:00</updated_at>
      <vapp_name>invapp</vapp_name>
      <hypervisor_id type="integer">28</hypervisor_id>
    </blueprint_template>
  </relation>
</relations>
</object>
</objects>

Where:

 objects - the blueprint template groups array with the following parameters:

 id - blueprint template group ID
 label - blueprint template group name
 parent_id - id of the target blueprint template group
 lft - left nested set identifier
 rgt - right nested set identifier
 depth - the depth of a given node (distance from this blueprint template group to the root)
 created_at - the date when the blueprint template group was created
 updated_at - the date when the blueprint template group was updated

 children - the array of child blueprint template groups

 relations - the array of blueprints assigned to the blueprint template groups with the following parameters:

 - id - relation ID
 - blueprint_template_id - blueprint template ID
 - blueprint_template_group_id - blueprint template group ID
 - created_at - the date when the blueprint template group was created
 - updated_at - the date when the blueprint template group was updated

 blueprint_template - an array of blueprint template parameters:

 - id - recipe ID
 - user_id - ID - the ID of a blueprint template owner
**Get Blueprint Template Group Details**

To get the list of blueprint template groups, use the following request:

GET /blueprint_template_groups.xml
GET /blueprint_template_groups.json

**XML Request example:**
```
```

**JSON Request example:**
```
```

**XML Output example**
```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint_template_group>
  <created_at type="datetime">2013-06-05T17:42:37+03:00</created_at>
  <depth type="integer">0</depth>
  <id type="integer">6</id>
  <label>in_blueprints</label>
  <lft type="integer">1</lft>
  <parent_id nil="true"/>
  <rgt type="integer">2</rgt>
  <updated_at type="datetime">2013-06-05T17:42:37+03:00</updated_at>
</blueprint_template_group>
```

Where:
- **created_at** - the date when the blueprint template group was created
- **updated_at** - the date when the blueprint template group was updated
- **depth** - the depth of a given node (distance from this blueprint template group to the root)
- **lft** - left nested set identifier
rgt - right nested set identifier
id - blueprint template group ID
label - blueprint template group name
parent_id - id of the target blueprint template group

19.3 Add Blueprint Template Group

To create new blueprint template group, use the following request:

POST /blueprint_template_groups.xml
POST /blueprint_template_groups.json

XML Request example:
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<blueprint_template_group><label>newgroup</label></blueprint_template_group>' --url http://onapp.test/blueprint_template_groups.xml

JSON Request example:
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"blueprint_template_group":{"label":"newgroup"}}' --url http://onapp.test/blueprint_template_groups.json

Where you have to specify the blueprint template group's label.

XML Response example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<blueprint_template_group>
  <created_at type="datetime">2013-06-07T10:49:39+00:00</created_at>
  <depth type="integer">0</depth>
  <id type="integer">9</id>
  <label>newgroup</label>
  <lft type="integer">5</lft>
  <parent_id nil="true"/>
  <rgt type="integer">6</rgt>
  <updated_at type="datetime">2013-06-07T10:49:39+00:00</updated_at>
</blueprint_template_group>
```

Where:
created_at – the date in the [YYYY][MM][DD][hh][mm][ss]Z format
id – the blueprint template group ID
label - the blueprint template group name
19.4 Edit Blueprint Template Group

To edit the blueprint template group, use the following request:

```
PUT /blueprint_template_groups/blueprint_template_group_id.xml
PUT /blueprint_template_groups/blueprint_template_group_id.json
```

**XML Request example:**
```
curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
  '<blueprint_template_group><label>newgroup</label></blueprint_template_group>' --url
  http://onapp.test/blueprint_template_groups/blueprint_template_group_id.xml
```

**JSON Request example:**
```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
  '{"blueprint_template_group":{"label":"newgroup"}}' --url
  http://onapp.test/blueprint_template_groups/blueprint_template_group_id.json
```

Where you can edit blueprint template group's label.

**XML Output example**
```
<objects type="array">
  <object>
    <id type="integer">2</id>
    <label>windows_blueprints</label>
    <parent_id nil="true"/>
    <lft type="integer">5</lft>
    <rgt type="integer">6</rgt>
    <depth type="integer">0</depth>
    <created_at type="datetime">2014-01-15T17:30:34+02:00</created_at>
    <updated_at type="datetime">2014-01-15T17:30:34+02:00</updated_at>
    <children type="array"/>
    <relations type="array"/>
  </object>
</objects>
```

Where:

- **objects** - the blueprint template groups array with the following parameters:
  - **id** - blueprint template group ID
  - **label** - blueprint template group name
  - **parent_id** - id of the target blueprint template group
  - **lft** - left nested set identifier
  - **rgt** - right nested set identifier
depth - the depth of a given node (distance from this blueprint template group to the root)
created_at - the date when the blueprint template group was created
updated_at - the date when the blueprint template group was updated
children - the array of child blueprint template groups
relations - the array of blueprints assigned to the blueprint template groups

19.5 Delete Blueprint Template Group

Use the following API request to delete a blueprint template group:

DELETE /blueprint_template_groups/blueprint_template_group_id.xml
DELETE /blueprint_template_groups/blueprint_template_group_id.json

XML Request example:
curl -i -X DELETE -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/blueprint_template_groups/blueprint_template_group_id.xml

JSON Request example:
curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/blueprint_template_groups/blueprint_template_group_id.json

Where you have to specify the blueprint template group's label.

XML Output example
<objects type="array">
<object>
  <id type="integer">2</id>
  <label>windows_blueprints</label>
  <parent_id nil="true"/>
  <lft type="integer">5</lft>
  <rgt type="integer">6</rgt>
  <depth type="integer">0</depth>
  <created_at type="datetime">2014-01-15T17:30:34+02:00</created_at>
  <updated_at type="datetime">2014-01-15T17:30:34+02:00</updated_at>
  <children type="array"/>
  <relations type="array"/>
</object>
</objects>

Where:
objects - the blueprint template groups array with the following parameters:
id - blueprint template group ID
label - blueprint template group name
parent_id - id of the target blueprint template group
lft - left nested set identifier
rgt - right nested set identifier
depth - the depth of a given node (distance from this blueprint template group to the root)
created_at - the date when the blueprint template group was created
updated_at - the date when the blueprint template group was updated
children - the array of child blueprint template groups
relations - the array of blueprints assigned to the blueprint template groups

19.6 Add Child Blueprint Template Group

To create new child blueprint template group, use the following request:

POST /blueprint_template_groups.xml
POST /blueprint_template_groups.json

XML Request example:

JSON Request example:
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"blueprint_template_group":{"label":"newgroup","parent_id":"1"}}' --url http://onapp.test/blueprint_template_groups.json

Where you have to specify the blueprint template group's label and ID of the parent group.

19.7 Get List of Blueprint Templates Attached to Blueprint Template Group

To get the list of blueprint template groups, use the following request:
WHITELIST IPS

EDIT WHITELISTED IP

GET /blueprint_template_groups/:blueprint_template_group_id/blueprint_template_group_relations.xml
GET /blueprint_template_groups/:blueprint_template_group_id/blueprint_template_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/blueprint_template_groups/:blueprint_template_group_id/blueprint_template_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/blueprint_template_groups/:blueprint_template_group_id/blueprint_template_group_relations.json

Where you have to specify the template group's ID in the URL.

XML Output example
<?xml version="1.0" encoding="UTF-8"?>
<blueprint_templates type="array">
  <blueprint_template>
    <built type="boolean">false</built>
    <created_at type="datetime">2013-06-07T11:26:53+00:00</created_at>
    <hypervisor_id type="integer">28</hypervisor_id>
    <id type="integer">16</id>
    <label>vmware_template_vapp</label>
    <updated_at type="datetime">2013-06-07T11:26:53+00:00</updated_at>
    <user_id type="integer">1</user_id>
    <vapp_name>invapp</vapp_name>
  </blueprint_template>
</blueprint_templates>

Where:
created_at - the date when the blueprint template was created in the [YYYY][MM][DD]T[hh][mm][ss]Z format
updated_at - the date when the blueprint template was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format
hypervisor_id - ID of a compute resource the vApp is located on
id - blueprint template ID
label - blueprint template name
user_id - ID of a user the blueprint template belongs to
vapp_name - name of the vApp this template is created from
19.8 Attach Blueprint Template to Group

Use the following request to attach blueprint template to the blueprint template group:

POST /blueprint_template_groups/:blueprint_template_group_id/blueprint_template_group_relations.xml

POST /blueprint_template_groups/:blueprint_template_group_id/blueprint_template_group_relations.json

**XML Request example**

```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass 
-d '<blueprint_template_group_relation><blueprint_template_id>1</blueprint_template_id></blueprint_template_group_relation>' --url http://onapp.test/blueprint_template_groups/:blueprint_template_group_id/blueprint_template_group_relations.xml
```

**JSON Request example**

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"blueprint_template_group_relation":{"blueprint_template_id":":1" }}' --url http://onapp.test/blueprint_template_groups/:blueprint_template_group_id/blueprint_template_group_relations.json
```

Where you have to specify the blueprint template ID and the blueprint template group ID.

19.9 Remove Blueprint Template from Blueprint Template Group

Use the following request to attach blueprint template to the blueprint template group:

DELETE /blueprint_template_groups/:id/blueprint_template_group_relations/:id.xml
DELETE /blueprint_template_groups/:id/blueprint_template_group_relations/:id.json

XML Request example
   curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass
   http://onapp.test/blueprint_template_groups/:id/blueprint_template_group_relations/:id.xml

JSON Request example
   curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass
   http://onapp.test/blueprint_template_groups/:id/blueprint_template_group_relations/:id.json

Where you have to specify the blueprint template ID and the blueprint template group ID in the URL.
20 BLUEPRINT TEMPLATES

Blueprint templates are VMware vApps images with operating system and network configuration settings that are used for blueprint creation.

20.1 Get List of Blueprint Templates

Use the following API call to get the list of blueprint templates:
GET /blueprint_templates.xml
GET /blueprint_templates.json

XML Request example

JSON Request example

XML Output example
<?xml version="1.0" encoding="UTF-8"?>
<blueprint_templates type="array">
<blueprint_template>
<created_at type="datetime">2013-06-01T09:09:00+00:00</created_at>
<hypervisor_id type="integer">28</hypervisor_id>
<id type="integer">1</id>
<label>vmware_template_vapp</label>
<updated_at type="datetime">2013-06-01T09:09:00+00:00</updated_at>
<user_id type="integer">1</user_id>
<vapp_name>invapp</vapp_name>
</blueprint_template>
</blueprint_templates>

Where:
created_at - the date when the blueprint template was created in the
[YYYY][MM][DD]T[hh][mm][ss]Z format
updated_at - the date when the blueprint template was updated in the
[YYYY][MM][DD]T[hh][mm][ss]Z format
hypervisor_id - ID of a compute resource the vApp is located on
id - blueprint template ID
label - blueprint template name
user_id - ID of a user the blueprint template belongs to
vapp_name - name of the vApp this template is created from

20.2 Get Blueprint Template Details

Use the following API call to get blueprint template details:
GET /blueprint_templates/:blueprint_template_id.xml
GET /blueprint_templates/:blueprint_template_id.json

XML Request example

JSON Request example

XML Output example
<?xml version="1.0" encoding="UTF-8"?>
<blueprint_template>
  <created_at type="datetime">2013-06-01T09:09:09+00:00</created_at>
  <hypervisor_id type="integer">28</hypervisor_id>
  <id type="integer">1</id>
  <label>vmware_template_vapp</label>
  <updated_at type="datetime">2013-06-01T09:09:09+00:00</updated_at>
  <user_id type="integer">1</user_id>
  <vapp_name>invapp</vapp_name>
</blueprint_template>

Where:
WHITELIST IPS - EDIT WHITELISTED IP

20.3 Add Blueprint Template

Use the following API call to create a blueprint template:

POST /blueprint_templates.xml
POST /blueprint_templates.json

**XML Request example**
```
curl -i -X POST -H 'Accept: application/xml' -H 'Content-Type: application/xml' -d
  '<blueprint_template><label>newlabel</label><hypervisor_id>28</hypervisor_id><vapp_name>vapp1</vapp_name></blueprint_template>' -u user:userpass --url
  http://onapp.test/blueprint_templates/import.xml
```

**JSON Request example**
```
curl -i -X POST -H 'Accept: application/json' -H 'Content-Type: application/json' -d
  '{"blueprint_template":{"label":"newlabel","hypervisor_id":"28","vapp_name":"vapp1"}}' -u user:userpass --url
  http://onapp.test/blueprint_templates/import.json
```

Where you have to specify:
- **label** - blueprint template label
- **hypervisor_id** - ID of a compute resource on which the vApps is located
- **vapp_name** - name of the vApp to create the template from

**XML Output example**
```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint_template>
  <built type="boolean">false</built>
</blueprint_template>
```
20.4 Edit Blueprint Template

Use the following API request to edit blueprint template:

PUT /blueprint_templates/:blueprint_template_id.xml
PUT /blueprint_templates/:blueprint_template_id.json

XML Request example

curl -i -X PUT -H 'Accept: application/xml' -H 'Content-type:application/xml' -d
  '<blueprint_template><label>newlabel</label></blueprint_template>'
-u user:userpass
http://onapp.test/blueprint_templates/:blueprint_template_id.xml

JSON Request example

curl -i -X PUT -H 'Accept: application/json' -H 'Content-type:application/json' -d
  '{"blueprint_template":{"label":"newlabel"}}' -u user:userpass --url
  http://onapp.test/blueprint_templates/:blueprint_template_id.json

Where you can edit the template's label.

XML Output example

<blueprint_template>
  <built type="boolean">true</built>
  <created_at type="datetime">2014-04-02T09:57:27+03:00</created_at>
  <hypervisor_id type="integer">70</hypervisor_id>
  <id type="integer">4</id>
  <label>linux_vapp</label>
  <updated_at type="datetime">2014-04-02T09:57:27+03:00</updated_at>
  <user_id type="integer">5</user_id>
  <vapp_name>lin_vapp</vapp_name>
</blueprint_template>

Where:
**WHITELIST IPS - EDIT WHITELISTED IP**

20.5 Delete Blueprint Template

Use the following API call to delete a blueprint template:

DELETE /blueprint_templates/:blueprint_template_id.xml
DELETE /blueprint_templates/:blueprint_template_id.json

**XML Request example**

```
```

**JSON Request example**

```
```

Where you have to specify ID of a blueprint template you want to remove.

**XML Response example**

```
<blueprint_templates type="array">
  <blueprint_template>
    <built type="boolean">true</built>
    <created_at type="datetime">2014-04-02T09:57:27+03:00</created_at>
    <hypervisor_id type="integer">70</hypervisor_id>
    <id type="integer">4</id>
    <label>linux_vapp</label>
    <updated_at type="datetime">2014-04-02T09:57:27+03:00</updated_at>
    <user_id type="integer">5</user_id>
    <vapp_name>lin_vapp</vapp_name>
  </blueprint_template>
</blueprint_templates>
```
Where:

- **built** - true if the template is built; otherwise false
- **created_at** - the date when the blueprint template was created
- **hypervisor_id** - the ID of the compute resource used by this blueprint template
- **id** - blueprint template ID
- **label** - blueprint template name
- **updated_at** - the date when the blueprint template was updated
- **user_id** - the ID of the user
- **vapp_name** - name of the vApp to create the template from
21 CATALOGS

This section provides the API calls you can use to manage catalogs imported from vCloud Director.

21.1 Get List of Catalogs

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

GET /catalogs.xml
GET /catalogs.json

XML Request example:

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

JSON Request example:

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

XML Output example:

```xml
<vcloud_catalogs type="array">
  <vcloud_catalog>
    <user_id>null</user_id>
    <hypervisor_id>4</hypervisor_id>
    <created_at>2016-02-01T11:43:52+00:00</created_at>
    <updated_at>2016-02-01T11:43:52+00:00</updated_at>
    <label>vn-onapp-public8</label>
    <published>true</published>
    <user_group_id>11</user_group_id>
    <identifier>a6d6d29a-e8eb-4869-a9af-53a5ec9b792c</identifier>
  </vcloud_catalog>
  ...
</vcloud_catalogs>
```

Where:
- `user_id` - the owner ID
- `hypervisor_id` - the ID 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
- `label` - the name of the catalog
- `published` - true if catalog is published
**WHITELIST IPS**

- **EDIT WHITELISTED IP**

---

user_group_id - the ID of the organisation, to which the catalog is assigned  
identifier - the identifier of the catalog  
id - the ID of the catalog

### 21.2 Get Catalog Details

To view the details of catalog, use the following request:

GET /catalogs/:id.xml  
GET /catalogs/:id.json

**XML Request example:**
```bash
curl -i -X GET -u user:userpass --url  
'Content-type: application/xml'
```

**JSON Request example:**
```bash
curl -i -X GET -u user:userpass --url  
http://onapp.test/catalogs/:id.json -H 'Accept: application/json' -  
H 'Content-type: application/json'
```

**XML Output example:**
```
<vcloud_catalog>  
  <user_id>null</user_id>  
  <hypervisor_id>4</hypervisor_id>  
  <created_at>2016-02-01T11:43:52+00:00</created_at>  
  <updated_at>2016-02-01T11:43:52+00:00</updated_at>  
  <label>vn-onapp-public8</label>  
  <published>true</published>  
  <user_group_id>11</user_group_id>  
  <identifier>a6d6d29a-e8eb-4869-a9af-53a5ec9b792c</identifier>  
  <id>8</id>  
</vcloud_catalog>
```

Where:  
user_id - the owner ID  
hypervisor_id - the ID 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  
label - the name of the catalog  
published - true if catalog is published  
user_group_id - the ID of the organisation, to which the catalog is assigned  
id - the identifier of the catalog  
in - the ID of the catalog
21.3 Create Catalog

To create a catalog, use the following request:

POST /catalogs.xml
POST /catalogs.json

**XML Request example:**
```
curl -i -X POST -u user:userpass --url
'Content-type: application/xml' -d
'&lt;vcloud_catalog&gt;&lt;user_group_id&gt;11&lt;/user_group_id&gt;&lt;data_store_id&gt;9&lt;/data_store_id&gt;&lt;vdc_id&gt;6&lt;/vdc_id&gt;&lt;label&gt;TestCatalog&lt;/label&gt;&lt;/vcloud_catalog&gt;
```

**JSON Request example:**
```
curl -i -X POST -u user:userpass --url
'Content-type: application/json' -d '{"vcloud_catalog": {"user_group_id": "11", "data_store_id": 9, "vdc_id": "6", "label": "TestCatalog"}}'
```

Where:
- **user_group_id** - the ID of the organisation, to which the catalog will be assigned
- **data_store_id** - the ID of the data store, to which the catalog will be assigned
- **vdc_id** - the ID of the resource pool, to which the catalog will be assigned
- **label** - the name of the catalog

21.4 Add vApp to Catalog

To add a vApp to catalog, use the following request:

POST /vapps/:vapp_id/conversion.xml
POST /vapps/:vapp_id/conversion.json

**XML Request example:**
```
curl -i -X POST -u user:userpass --url
'Content-type: application/xml' -d ' &lt;vcloud-vapp-template&gt;&lt;catalog&gt;1&lt;/catalog&gt;&lt;overwrite-catalog-item&gt;1&lt;/overwrite-catalog-item&gt;&lt;label&gt;vApp_system_111&lt;/label&gt;&lt;description&gt;test&lt;/description&gt;&lt;target-vapp-template&gt;1&lt;/target-vapp-template&gt;&lt;/vcloud-vapp-template&gt;'
```

**JSON Request example:**
```
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H
'Content-Type: application/json'
```
http://onapp.test/vapps/:vapp_id/conversion.json -d
'{"vcloud_vapp_template":{"catalog":"1","overwrite_catalog_item":"1","label":"vApp_system_111","description":"test","target_vapp_template":"1"}}'}

Where:
catalog - choose the catalog to which the vApp will be added.
overwrite_catalog_item - set "1" to save this vApp as template instead of another vApp template, otherwise set "0".
label - specify the name of the vApp. This parameter is applicable only when the overwrite_catalog_item parameter is set to "0".
description - add the appropriate vApp description.
target_vapp_template - choose the appropriate vApp template, which will be replaced. This parameter is applicable only when the overwrite_catalog_item parameter is set to "1".

21.5 Delete Catalog

To delete a catalog, use the following request:
DELETE /catalogs/:id.xml
DELETE /catalogs/:id.xml
XML Request example:
curl -i -X DELETE -u user:userpass --url
'Content-type: application/xml'

JSON Request example:
curl -i -X DELETE -u user:userpass --url
http://onapp.test/catalogs/:id.json -H 'Accept: application/json' -
H 'Content-type: application/json'

21.6 Get List of vApp Templates

To view the list of vApp templates, use the following request:
GET/catalogs/:id/vapp_templates.xml
GET/catalogs/:id/vapp_templates.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
<vcloud_vapp_templates type="array">
    <vcloud_vapp_template>
        <virtual_machines>
            <names type="array"></names>
        </virtual_machines>
        <created_at>2016-02-01T11:44:24+00:00</created_at>
        <updated_at>2016-02-01T11:44:24+00:00</updated_at>
        <label>Centos66_net</label>
        <catalog_item_id>15</catalog_item_id>
        <identifier>vappTemplate-51e0fc11-5d6c-46a9-a96c-add3d5b6edca</identifier>
        <id>12</id>
    </vcloud_vapp_template>
    <vcloud_vapp_template>...
</vcloud_vapp_templates>
```

Where:
- **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
- **label** - the name of the vApp template
- **catalog_item_id** - ID of the catalog
- **identifier** - the identifier of the vApp template
- **id** - ID of the vApp template

## 21.7 Get List of Media Files

To view the list of Media files, use the following request:

```sh
get/catalogs/:id/media.xml
get/catalogs/:id/media.json
```

**XML Request example:**
```sh
```

**JSON Request example:**
```sh
```

**XML Output example:**
```xml
<vcloud_media type="array">
    <vcloud_media>
```

```
```
<status>1</status>
<user_id>null</user_id>
<description>test</description>
<data_store_id>2</data_store_id>
<image_type>iso</image_type>
<created_at>2016-02-04T14:19:26+02:00</created_at>
<updated_at>2016-02-04T14:19:26+02:00</updated_at>
<label>Kostya000</label>
<catalog_item_id>3</catalog_item_id>
<identifier>6677d99d-a3af-40d0-aa4a-907b41ab2559</identifier>
<vdc_id>1</vdc_id>
<size>414187520</size>
</vcloud_media>
</vcloud_media>

Where:
status - the status of media file
user_id - owner ID
description - the media file description
data_store_id - the ID of the data store to which the media file is assigned
image_type - the type of the media file
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
label - the name of the media file
catalog_item_id - ID of the catalog
identifier - the identifier of the media file
vdc_id - the ID of the resource pool
id - the ID of the media file
size - the size of media file

21.8 Add File to Media Library

To add a file to media library, take the following steps:
1. Upload file to the Control Panel.
2. Add file to media library of a specific catalog.

21.8.1 File uploading to the Control Panel

To upload file to the Control Panel, use the following request:
POST /catalogs/upload_files

Request example:

Where:
$vcloud_media[folder]$ - path, where the media file is stored

21.8.2 Adding file to media library

To add a file to media library, use the following request:

**POST /catalogs/:id/media.xml**
**POST /catalogs/:id/media.json**

**XML Request example:**

**JSON Request example:**
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json' http://onapp.test/catalogs/:id/media.json -d '{"vcloud_media":{"label":"Media","folder":"path"}}'

Where:
$label*$ - the name of the media file
$folder*$ - path, where the media file is situated (the same folder, as indicated during file uploading to the Control Panel)

21.9 Create vApp Template (Add to Catalog)

To create a vApp template, use the following request:

**POST /catalogs/:id/vapp_templates.xml**
**POST /catalogs/:id/vapp_templates.json**

**XML Request example:**
JSON Request example:
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json'
http://onapp.test/catalogs/:id/vapp_templates.json -d '{"vcloud_vapp_template":{"label":"Template","ovf_url":"url"}}'

Where:
label - the name of the vApp template
ovf_url - the link to the vApp template
or
folder - path, where the vApp template is situated

21.10 Delete vApp Template

To delete a vApp template, use the following request:
DELETE /catalogs/:id/vapp_templates/:id.xml
DELETE /catalogs/:id/vapp_templates/:id.xml

XML Request example:
curl -i -X DELETE -u user:userpass -H 'Accept: application/xml' -H 'Content-Type: application/xml'
http://onapp.test/catalogs/:id/vapp_templates/:id.xml

JSON Request example:
curl -i -X DELETE -u user:userpass -H 'Accept: application/json' -H 'Content-Type: application/json'
http://onapp.test/catalogs/:id/vapp_templates/:id.json
22 CDN ACCELERATOR

This chapter provides requests for accelerators.

22.1 Get List of Accelerators

To view all accelerators in the cloud with their details, use the following request:
GET /accelerators.xml
GET /accelerators.json
XML Request example:
curl -i -X GET -u user:userpass http://onapp.test/accelerators.xml

JSON Request example:
curl -i -X GET -u user:userpass http://onapp.test/accelerators.json

XML Output example
<accelerators type="array">
<accelerator>
<admin_note nil="true"/>
<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_threads nil="true"/>
<cpu_units type="integer">10</cpu_units>
<cpus type="integer">1</cpus>
<created_at type="datetime">2015-09-16T14:39:03:00</created_at>
<customer_network_id nil="true"/>
<deleted_at nil="true"/>
<enable_autoscale nil="true"/>
<enable_monitor type="boolean">false</enable_monitor>
<firewall_notrack type="boolean">true</firewall_notrack>
<hot_add_cpu nil="true"/>
<hot_add_memory nil="true"/>
<hypervisor_id type="integer">1</hypervisor_id>
<id type="integer">323</id>
<identifier>g8u26b0gw5sr11</identifier>
<initial_root_password>5xkPnPToTv4J</initial_root_password>
<initial_root_password_encrypted>
type="integer">1095630</monthly_bandwidth_used>
<total_disk_size type="integer">20</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>
<edge_status>Active</edge_status>
<cdn_reference type="integer">410772127</cdn_reference>
</accelerator>
</accelerators>

Explanation of the data returned:

admin_note – an optional reminder for this accelerator created by an administrator
allowed_hot_migrate – true if hot migration is allowed; otherwise false
allowed_swap – true if swap is allowed; otherwise false
booted – true if the accelerator is booted; otherwise false
built – true if the accelerator is built; otherwise false
cores_per_socket – the number of cores per socket for accelerator
cpu_shares – the CPU priority percentage
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
cpus – number of CPU cores allocated to this accelerator
created_at – the date when the accelerator was created in the
YYYY[MM][DD][hh][mm][ss]Z format
customer_network_id – ID of a customer network
deleted_at – time when the accelerator was deleted
enable_autoscale – false; not available for accelerators
enable_monitis – deprecated attribute; will be removed in upcoming release
firewall_notrack – true if the NOTRACK rule is set in iptables
hot_add_cpu – false; not available for accelerators
hot_add_memory – false; not available for accelerators
hypervisor_id – the ID of the compute resource, on which the accelerator is deployed
id – the accelerator ID in OnApp CP database
identifier – the accelerator identifier
initial_root_password – the accelerator root password
initial_root_password_encrypted – true, if the accelerator root password is
encrypted, otherwise false
instance_package_id – false; not available for accelerators
iso_id - false; not available for accelerators
label - an arbitrary name of the accelerator
local_remote_access_ip_address - the IP address used for console access
local_remote_access_port - the port ID used for console access
locked - true if locked; otherwise false
memory - the amount of RAM resources allocated to this accelerator
min_disk_size - minimum disk space required by the template
note - an optional reminder for this accelerator made by a user account
operating_system - type of operating system
operating_system_distro - the distribution of the operating system
preferred_hvs - the array of preferable compute resources based on compute zone that meet some accelerator configuration settings
recovery_mode - true if the accelerator is booted in the recovery mode; otherwise false
remote_access_password - the password for remote access
service_password - service account password
state - deprecated attribute; will be removed in upcoming release
strict_virtual_machine_id - the ID of a virtual server (or edge server) that will never reside on the same compute resource with this accelerator
suspended - true if suspended; otherwise false
template_id - the ID of the template, on which the accelerator is based
template_label - label of the template on which the accelerator is based
time_zone - the time zone set for the accelerator
updated_at - the date when the accelerator was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
user_id - the ID of the user, who is the accelerator owner
vip - true if the accelerator has VIP status for migration; otherwise false
xen_id - the accelerator ID set by the virtualization engine
ip_addresses - an array of assigned IP addresses with their details assigned to this accelerator:
  • address - IP address
  • 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
  • network_address - IP 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 - 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
22.2 Get Accelerator Details

To view the accelerator details:
GET /accelerators/:id.xml
GET /accelerators/:id.json

XML Request example
```bash
curl -i -X GET -u user:userpass
http://onapp.test/accelerators/:id.xml
```

JSON Request example:
```bash
curl -i -X GET -u user:userpass
http://onapp.test/accelerators/:id.json
```

XML Output example
```xml
<accelerator>
  <admin_note nil="true"/>
  <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>
  <cpusockets nil="true"/>  
  <cpu_threads nil="true"/>
  <cpu_units type="integer">10</cpu_units>
  <cpus type="integer">1</cpus>
  <created_at type="datetime">2015-09-16T14:41:39+03:00</created_at>
  <customer_network_id nil="true"/>
  <deleted_at nil="true"/>
  <enable_autoscale nil="true"/>
  <enable_monitis type="boolean">false</enable_monitis>
</accelerator>
```
<firewall_notrack type="boolean">true</firewall_notrack>
<hot_add_cpu nil="true"/>
<hot_add_memory nil="true"/>
<hypervisor_id type="integer">1</hypervisor_id>
<identifier>g8u26b0gw5sr11</identifier>
<initial_root_password>5xkPnPToIv4J</initial_root_password>
<initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
<instance_package_id nil="true"/>
<iso_id nil="true"/>
<label>natalia</label>
<local_remote_access_ip_address>69.168.237.15</local_remote_access_ip_address>
<local_remote_access_port type="integer">5900</local_remote_access_port>
<locked type="boolean">false</locked>
<memory type="integer">2048</memory>
<min_disk_size type="integer">20</min_disk_size>
<note nil="true"/>
<operating_system>linux</operating_system>
<operating_system_distro>ubuntu</operating_system_distro>
<preferred_hvs type="array"/>
<recovery_mode type="boolean">false</recovery_mode>
<remote_access_password>Y6eHcWfZsd8V</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">23</template_id>
<template_label>OnApp CDN Appliance</template_label>
<time_zone>Athens</time_zone>
<updated_at type="datetime">2015-09-23T17:58:03+03:00</updated_at>
=user_id type="integer">79</user_id>
<vip nil="true"/>
<xen_id type="integer">92</xen_id>
<ip_addresses type="array">
<ip_address>
<address>194.44.20.82</address>
broadcast>194.44.20.255</broadcast>
<created_at type="datetime">2013-11-14T15:48:37+03:00</created_at>
<customer_network_id nil="true"/>
<disallowed_primary type="boolean">false</disallowed_primary>
<gateway>194.44.20.1</gateway>
<hypervisor_id nil="true"/>
<id type="integer">3</id>
<ip_address_pool_id nil="true"/>
<network_address>194.44.20.0</network_address>
<network_id type="integer">1</network_id>
<pxe type="boolean">false</pxe>
<updated_at type="datetime">2015-09-14T12:16:16+03:00</updated_at>
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>255.255.255.0</netmask>
</ip_address>
</ip_addresses>
<monthly_bandwidth_used type="integer">938418</monthly_bandwidth_used>
<total_disk_size type="integer">20</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>
<edge_status>Active</edge_status>
<cdn_reference type="integer">410772127</cdn_reference>
</accelerator>

Explanation of the data returned:

- **admin_note** - an optional reminder for this accelerator created by an administrator
- **allowed_hot_migrate** - true if hot migration is allowed; otherwise false
- **allowed_swap** - true if swap is allowed; otherwise false
- **booted** - true if the accelerator is booted; otherwise false
- **built** - true if the accelerator is built; otherwise false
- **cores_per_socket** - the number of cores per socket for accelerator
- **cpu_shares** - the CPU priority percentage
- **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
- **cpus** - number of CPU cores allocated to this accelerator
- **created_at** - the date when the accelerator was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- **customer_network_id** - ID of a customer network
- **deleted_at** - time when the accelerator was deleted
- **enable_autoscale** - false; not available for accelerators
- **enable_monitis** - deprecated attribute; will be removed in upcoming release
- **firewall_notrack** - true if the NOTRACK rule is set in iptables
- **hot_add_cpu** - false; not available for accelerators
- **hot_add_memory** - false; not available for accelerators
- **hypervisor_id** - the ID of the compute resource, on which the accelerator is deployed
id - the accelerator ID in OnApp CP database
identifier - the accelerator identifier
initial_root_password - the accelerator root password
initial_root_password_encrypted - true, if the accelerator root password is encrypted, otherwise false
instance_package_id - false; not available for accelerators
iso_id - false; not available for accelerators
label - an arbitrary name of the accelerator
local_remote_access_ip_address - the IP address used for console access
local_remote_access_port - the port ID used for console access
locked - true if locked; otherwise false
memory - the amount of RAM resources allocated to this accelerator
min_disk_size - minimum disk space required by the template
note - an optional reminder for this accelerator made by a user account
operating_system - type of operating system
operating_system_distro - the distribution of the operating system
preferred_hvs - the array of preferable compute resources based on compute zone that meet some accelerator configuration settings
recovery_mode - true if the accelerator is booted in the recovery mode; otherwise false
remote_access_password - the password for remote access
service_password - service account password
state - deprecated attribute; will be removed in upcoming release
strict_virtual_machine_id - the ID of a virtual server (or edge server) that will never reside on the same compute resource with this accelerator
suspended - true if suspended; otherwise false
template_id - the ID of the template, on which the accelerator is based
template_label - label of the template on which the accelerator is based
updated_at - the date when the accelerator was updated in the [YYYY][MM][DD]T[hh][mm][ss]Z format
user_id - the ID of the user, who is the accelerator owner
vip - true if the accelerator has VIP status for migration; otherwise false
xen_id - the accelerator ID set by the virtualization engine
ip_addresses - an array of assigned IP addresses with their details assigned to this accelerator:
  • address - IP address
  • 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]T[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
  • network_address - IP 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** - 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

**monthly_bandwidth_used** - accelerator monthly bandwidth in KB

**total_disk_size** - total disk space in GB of primary and swap disks

**price_per_hour** - accelerator's price per hour

**price_per_hour_powered_off** - price per hour when accelerator is powered off

**support_incremental_backups** - 1, if accelerator 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**

**edge_status** - the CDN server status

**cdn_reference** - the identifier in database

### 22.3 Add Accelerator

To create an accelerator, use the following API call:

POST /accelerators.xml

POST /accelerators.json

Below you can find requirements for Accelerator creation:

- **Minimum**: 4 cores, 4GB RAM and 100GB disks
- **Recommended**: 8 cores, 16 GB RAM and 1TB disks
- SSD recommended to avoid slowing down access

**XML Request example**

curl -i -X POST -d

'<?xml version="1.0" encoding="UTF-8"?>
<accelerator>
  <label>test</label>
  <cpus>1</cpus>
  <data_store_group_primary_id>2</data_store_group_primary_id>
  <primary_network_group_id>3</primary_network_group_id>
  <cpu_shares>1</cpu_shares>
  <memory>512</memory>
  <required_virtual_machine_build>1</required_virtual_machine_build>
  <hypervisor_group_id>1</hypervisor_group_id>
  <hypervisor_id>1</hypervisor_id>
  <required_ip_address_assignment>1</required_ip_address_assignment>
  <primary_disk_size>5</primary_disk_size>
  <rate_limit>0</rate_limit>
</accelerator>' -u user:password

JSON Request example

curl -i -X POST -d
'{"accelerator":{"label":"test","cpus":1,"data_store_group_primary_id":2,"primary_network_group_id":3,"cpu_shares":1,"memory":512,"required_virtual_machine_build":1,"hypervisor_group_id":1,"hypervisor_id":1,"required_ip_address_assignment":1,"primary_disk_size":5,"rate_limit":0}}' -u user:userpass

Where:

- **label** – a unique name of your accelerator. The label can consist of letters [A-Za-z], digits [0-9], dash [-], lower dash [ _ ], space character [ ], at sign [@], round brackets [(]), slashes [/], comma [,] and dot [.]. You can use both lower- and uppercase letters. The label should begin with an alphanumeric character or lower dash [ _ ]
- **hypervisor_id** - indicate the ID of the compute resource, on which the accelerator will be deployed
- **hypervisor_group_id** - indicate the compute zone ID
- **cpus** - the amount of CPU cores allocated to this accelerator
- **cpu_shares** - the percentage of allocated CPU priority resource
- **memory** - the amount of RAM, which you want to allocate to this accelerator
- **primary_disk_size** - the size in GB of the primary disk
- **rate_limit** - the port speed
- **data_store_group_primary_id** - specify the ID of a data store zone, where you want to locate the disk of your accelerator. If not specified - the system will select the data store zone with higher available capacity
- **primary_network_group_id** - indicate the network zone ID
- **required_virtual_machine_build** - set "1" to build the accelerator automatically after creation. Otherwise set "0"
- **required_ip_address_assignment** - set "1" if you want IP address to be assigned automatically after creation. Otherwise set "0"

22.4 Edit Accelerator

To change the accelerator label, resource allocation:

PUT /accelerators/:id.xml
PUT /accelerators/:id.json

XML Request example

curl -i -X PUT -d
  '〈accelerator〉<label>test</label><cpus>1</cpus><cpu_shares>10</cpu_shares><memory>512</memory></accelerator>'' -u onapp.test
22.5 Reboot Accelerator

To reboot the accelerator:

POST /accelerators/:accelerator_id/reboot.xml
POST /accelerators/:accelerator_id/reboot.json

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/accelerators/:accelerator_id/reboot.xml -H
 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example

curl -i -X POST -u user:userpass
http://onapp.test/accelerators/:accelerator_id/reboot.json -H
 'Accept: application/json' -H 'Content-type: application/json'

22.6 Start up Accelerator

POST /accelerators/:accelerator_id/startup.xml
POST /accelerators/:accelerator_id/startup.json

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/accelerators/:accelerator_id/startup.xml -H
 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example
22.7 Shut down Accelerator

To terminate the edge server gracefully:

POST /accelerators/:accelerator_id/shutdown.xml
POST /accelerators/:accelerator_id/shutdown.json

**XML Request example**

```
curl -i -X POST -u user:userpass
http://onapp.test/accelerators/:accelerator_id/shutdown.xml -H
'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request example**

```
curl -i -X POST -u user:userpass
http://onapp.test/accelerators/:accelerator_id/shutdown.json -H
'Accept: application/json' -H 'Content-type: application/json'
```

22.8 Suspend Accelerator

POST /accelerators/:accelerator_id/suspend.xml
POST /accelerators/:accelerator_id/suspend.json

**XML Request example**

```
curl -i -X POST -u user:userpass
http://onapp.test/accelerators/:accelerator_id/suspend.xml -H
'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request example**

```
curl -i -X POST -u user:userpass
http://onapp.test/accelerators/:accelerator_id/suspend.json -H
'Accept: application/json' -H 'Content-type: application/json'
```

To unsuspend the accelerator, run the request again.

22.9 Rebuild Accelerator

To rebuild (or build manually) the accelerator, use the following request:

POST /accelerators/:accelerator_id/build.xml
POST /accelerators/:accelerator_id/build.json
**22.10 Migrate Accelerator**

To migrate an accelerator to another compute resource, use the following request:

POST /accelerators/:accelerator_id/migrate.xml
POST /accelerators/:accelerator_id/migrate.json

Currently, accelerators support only cold migration.

**XML Request example**

```
curl -i -X POST -u user:userpass
  http://onapp.test/accelerators/:accelerator_id/build.xml
```

**JSON Request example**

```
curl -i -X POST -u user:userpass
  http://onapp.test/accelerators/:accelerator_id/build.json
```

Where:
- `destination` - the ID of a target compute resource, to which you migrate the accelerator
- `cold_migrate_on_rollback` - set 1 if you wish to switch to a cold migration if hot migration fails, otherwise set 0.

**22.11 Delete Accelerator**

DELETE /accelerators/:id.xml
DELETE /accelerators/:id.json

**XML Request example**

```
curl -i -X DELETE -u user:userpass
  http://onapp.test/accelerators/accelerator_id.xml
  --url
  http://onapp.test/accelerators/accelerator_id.xml
```

**JSON Request example**

```
curl -i -X DELETE -u user:userpass
  http://onapp.test/accelerators/accelerator_id.json
  --url
  http://onapp.test/accelerators/accelerator_id.json
```
**22.12 Unlock Accelerator**

To unlock the accelerator:

**POST /accelerators/:accelerator_id/unlock.xml**
**POST /accelerators/:accelerator_id/unlock.json**

**XML Request example**

```
curl -i -X POST -u user:userpass  
```

**JSON Request example**

```
curl -i -X POST -u user:userpass  
```

If there are accelerated virtual servers in the cloud, these VSs will still be billed for acceleration even if you delete the accelerator.

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

**22.13 Segregate Accelerator**

To segregate an accelerator (that is, instruct it never to reside on the same compute resource with another accelerator), use the following method:

**POST /accelerators/:accelerator_id/strict_vm.xml**
**POST /accelerators/:accelerator_id/strict_vm.json**

**XML Request example**

```
curl -i -X POST -u user:userpass  
-d '<?xml version="1.0" encoding="UTF-8"?>
<virtual_machine>
<strict_virtual_machine_id>bb6oa3eqdzpcgl</strict_virtual_machine_id>
</virtual_machine>' --url http://onapp.test/accelerators/:accelerator_id/strict_vm.xml
```

**JSON Request example**

```
curl -i -X POST -u user:userpass -d
```

**JSON Request example**

```
curl -i -X POST -u user:userpass  
```
22.14 Change Accelerator Owner

Use the following request to reassign an accelerator to another user:

POST /accelerators/:accelerator_id/change_owner.xml
POST /accelerators/:accelerator_id/change_owner.json

XML Request example

JSON Request example
*curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"user_id":1}' --url http://onapp.test/accelerators/:accelerator_id/change_owner.json*

Where:
user_id * - input ID of a new owner

22.15 Accelerator Network Interfaces

Here is the list of API calls for managing accelerators' network interfaces. Accelerators' network interfaces have the same attributes as network interfaces of virtual servers.

To get the list of network interfaces allocated to this particular accelerator:
GET /accelerators/:accelerator_id/network_interfaces.xml
GET /accelerators/:accelerator_id/network_interfaces.json

To get a particular network interface details:
GET /accelerators/:accelerator_id/network_interfaces/:id.xml
GET /accelerators/:accelerator_id/network_interfaces/:id.json

To edit network interface details:
PUT /accelerators/:accelerator_id/network_interfaces/:id.xml
PUT /accelerators/:accelerator_id/network_interfaces/:id.json

To add a new network interface:
POST /accelerators/:accelerator_id/network_interfaces.xml
POST /accelerators/:accelerator_id/network_interfaces.json

To delete a network interface from the accelerator:
DELETE /accelerators/:accelerator_id/network_interfaces/:id.xml
DELETE /accelerators/:accelerator_id/network_interfaces/:id.json

XML Output example
    <network_interface>
      <connected nil="true"/>
      <created_at type="datetime">2015-09-25T14:33:13+03:00</created_at>
      <default_firewall_rule>ACCEPT</default_firewall_rule>
      <id type="integer">372</id>
      <identifier>yekx0libarssan</identifier>
      <label>eth0</label>
      <mac_address>00:16:3e:81:42:83</mac_address>
      <network_join_id type="integer">5</network_join_id>
      <primary>true</primary>
      <rate_limit type="integer">1</rate_limit>
      <updated_at type="datetime">2015-09-25T14:33:13+03:00</updated_at>
      <usage nil="true"/>
      <usage_last_reset_at nil="true"/>
      <usage_month_rolled_at nil="true"/>
      <virtual_machine_id type="integer">359</virtual_machine_id>
    </network_interface>

Where:
  label - network interface name
  created_at - the timestamp in the database when this network interface was created
  default_firewall_rule - set default firewall rule for the particular network interface - either DROP or ACCEPT
  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 an accelerator to which this network interface is attached
  connected - not relevant to accelerators
  usage - not relevant to accelerators
  usage_last_reset_at - not relevant to accelerators
  usage_month_rolled_at - not relevant to accelerators
22.16 Accelerator IP Address Joins

An IP address allocated to an accelerator is an IP address join. Use the following methods to view, assign and delete IP address joins of your accelerators.

To get the list of IP address assignments for a particular accelerator:
GET /accelerators/:accelerator_id/ip_addresses.xml
GET /accelerators/:accelerator_id/ip_addresses.json

XML Request Example

Json Request Example

To assign an IP Address to an accelerator:
POST /accelerators/:accelerator_id/ip_addresses.xml
POST /accelerators/:accelerator_id/ip_addresses.json

XML Request Example

Json Request Example

XML Response Example
Status: 201
Content-Length: 1065
Connection: close
Content-Type: application/xml; charset=utf-8

<?xml version="1.0" encoding="UTF-8"?>
<ip_address_join>
  <created_at type="datetime">2013-10-31T13:04:05+03:00</created_at>
  <id type="integer">173</id>
</ip_address_join>
<ip_address_id type="integer">7</ip_address_id>
<network_interface_id type="integer">131</network_interface_id>
<updated_at type="datetime">2013-10-31T13:04:05+03:00</updated_at>

<ip_address>
  <address>1.1.1.3</address>
  <broadcast>1.1.1.255</broadcast>
  <created_at type="datetime">2013-08-07T13:29:09+03:00</created_at>
  <customer_network_id nil="true"/>
  <disallowed_primary type="boolean">false</disallowed_primary>
  <gateway>1.1.1</gateway>
  <hypervisor_id nil="true"/>
  <id type="integer">7</id>
  <ip_address_pool_id nil="true"/>
  <network_address>1.1.1.0</network_address>
  <network_id type="integer">1</network_id>
  <pxe type="boolean">false</pxe>

  <updated_at type="datetime">2013-08-07T13:29:09+03:00</updated_at>
  <user_id nil="true"/>
  <free type="boolean">false</free>
  <netmask>255.255.255.0</netmask>
</ip_address>
</ip_address_join>

Where:
created_at - the date when the record was created in DB
id - the IP address join ID
ip_address_id - the IP address ID
network_interface_id - the network interface ID
updated_at - the date when the record was updated in DB
ip_address - the array of IP address details
address - the IP address
broadcast - a logical address at which all devices connected to a multiple-access communications network are enabled to receive datagrams.
customer_network_id - the ID of the customer network
disallowed_primary - true if this address is not set as primary (for VS build), otherwise false
gateway - gateway address
hypervisor_id - the ID of the compute resource
ip_address_pool_id - the ID of the IP address pool to which this join belongs
network_address - the address of a VLAN network address that will be associated with this IP address pool
**network_id** - the ID of the network

**pxe** - true, if this address can be used for cloudbooting a compute resource

**free** - true if free, otherwise false

**netmask** - netmask for the IP address

**To delete an IP address assignment from a particular accelerator:**

DELETE /accelerators/:accelerator_id/ip_addresses/:id.xml

DELETE /accelerators/:accelerator_id/ip_addresses/:id.json

**XML Request Example**


**Json Request Example**


**Where:**

**data_store_id** - the ID of the data store, which is attached to the compute resource

**hypervisor_id** - reserved parameter

**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

---

### 22.17 View Accelerator Disks

To view the accelerator disks:

GET /accelerators/:accelerator_id/disks.xml

GET /accelerators/:accelerator_id/disks.json

**XML Output example**

```xml
<disks type="array">
  <disk>
    <mounted>true</mounted>
    <built type="boolean">true</built>
    <burst_bw type="integer">1000</burst_bw>
    <created_at type="datetime">2015-09-25T14:33:13+03:00</created_at>
    <data_store_id type="integer">4</data_store_id>
    <disk_size type="integer">20</disk_size>
    <disk_vm_number type="integer">1</disk_vm_number>
    <file_system type="symbol">ext3</file_system>
    <id type="integer">460</id>
    <identifier>pd60674pgnqfx4</identifier>
  </disk>
</disks>
```
<iqn nil="true"/>
<is_swap type="boolean">false</is_swap>
<label>Disk#460</label>
<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">2015-09-27T19:26:54+03:00</updated_at>
<virtual_machine_id type="integer">359</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 accelerator using this disk
volume_id - data store ID
has_autobackups - true if the disk has automatic backups set up, otherwise false
SolidFire - related parameters (irrelevant for accelerators)
iqn
burst_bw
max_bw
22.18 Rebuild Network for Accelerator

It is required to rebuild network after any changes on IP address joins or network interfaces. To rebuild network, use the following request:

POST /accelerators/:accelerator_id/rebuild_network.xml
POST /accelerators/:accelerator_id/rebuild_network.json

XML Request example

```plaintext
curl -X POST -u user:userpass --url
'http://onapp.test/accelerators/:accelerator_id/rebuild_network.xml
?force=1&shutdown_type=hard&required_startup=1' -H 'Accept:
application/xml' -H 'Content-type: application/xml'
```

JSON Request example

```plaintext
curl -X POST -u user:userpass --url
'http://onapp.test/accelerators/:accelerator_id/rebuild_network.json
?force=1&shutdown_type=hard&required_startup=1' -H 'Accept:
application/json' -H 'Content-type: application/json'
```

Where:

- `accelerator_id` - ID of the accelerator
- `shutdown_type` - type of the accelerator shutdown: hard, graceful or soft
- `required_startup` - set 1 to start up the accelerator automatically after build, otherwise set 0

22.19 Get Accelerator CPU Usage Statistics

To view CPU usage statistics of an accelerator, run:

GET /accelerators/:accelerator_id/cpu_usage.xml
GET /accelerators/:accelerator_id/cpu_usage.json

Define a shorter period by setting Start and End time in the API call:


XML Request example:

```plaintext
curl -i GET -u user:userpass --url
http://onapp.test/accelerators/:accelerator_id/cpu_usage.xml
```
XML Request example:

```bash
curl -i GET -u user:userpass --url
http://onapp.test/accelerators/:accelerator_id/cpu_usage.json
```

Where you have to specify the accelerator ID.
CDN edge groups are groups of edge servers – your own, and those you subscribe to from the CDN marketplace. They are usually grouped by location, so they represent a pool of servers for a given geographical area. Once you have created an edge group containing edge servers in specific locations, you can then assign the group (or groups) to a specific CDN resource. You need to associate CDN Edge groups with billing plans to make them available for users.

PLEASE NOTE: Starting from the OnApp Cloud v3.0, CDN is enabled automatically after adding the first DNS record or CDN resource.

### 23.1 Get List of CDN Edge Groups

To view CDN edge groups available in the cloud:

GET /edge_groups.xml
GET /edge_groups.json

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<edge_groups type="array">
  <edge_group>
    <label>tredty</label>
    <created_at type="datetime">2011-10-11T12:58:40Z</created_at>
    <updated_at type="datetime">2011-10-11T12:58:40Z</updated_at>
    <id type="integer">1</id>
  </edge_group>
...
  <edge_group></edge_group>
...
</edge_groups>
```

**Where:**
- *label* – the edge group label
- *id* – the group id in the database
23.2 Get List of Available CDN Edge Groups

To view the list of all the edge groups and their locations, which are available to create CDN resources on, use the following request:

GET /cdn_resources/available_edge_groups.xml
GET /cdn_resources/available_edge_groups.json

The list of available edge groups is defined by the billing plan to which a user is assigned.

XML Output example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<edge_groups type="array">
  <edge_group>
    <label>eg01</label>
    <edge_group_locations type="array">
      <edge_group_location>
        <city>dallas</city>
        <price type="decimal">0.7</price>
        <created_at type="datetime">2012-03-01T11:16:10+02:00</created_at>
        <country>US</country>
        <aflexi_location_id type="integer">147</aflexi_location_id>
        <updated_at type="datetime">2012-03-01T11:16:10+02:00</updated_at>
      </edge_group_location>
    </edge_group_locations>
    <created_at type="datetime">2012-03-01T11:09:28+02:00</created_at>
    <updated_at type="datetime">2012-03-01T11:09:28+02:00</updated_at>
    <id type="integer">35</id>
    <operator>WK</operator>
  </edge_group>
</edge_groups>
```

Where:
- `edge_groups` - the array of edge groups with their locations available for a user to create a CDN resource on.
- `edge_group` - the particular edge group details:
  - `label` - the edge group label
  - `edge_group_locations` - the array of locations assigned to this group:
    - `edge_group_location` - the list of details for a particular edge group
      - `city` - the city where the edge server is located
      - `price` - price per GB of sold excess bandwidth
      - `created_at` - the date when the record was created in DB
      - `country` - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format
      - `aflexi_location_id` - the ID of this location in Aflexi database
updated_at – the date when the record was updated in DB
id – the location ID
operator – the location operator
edge_group_id – the ID of the edge group to which this location is assigned

23.3 Get CDN Edge Group Details

To view the edge group details, use the following request:
GET /edge_groups/:id.xml
GET /edge_groups/:id.json

XML Request example
  curl -i -X GET -u user:userpass
  http://onapp.test/edge_groups/:edge_group_id.xml?available_locations=true
  -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example
  curl -i -X GET -u user:userpass
  http://onapp.test/edge_groups/:edge_group_id.json -d
  '{"available_locations":true}' -H 'Accept: application/json' -H 'Content-type:application/json'

Where:
available_locations - set true to view the list of locations available to this edge group; set false to view only the list of assigned locations.

XML Output example
  <?xml version="1.0" encoding="UTF-8"?>
  <edge_group>
    <created_at type="datetime">2012-04-18T11:33:01+00:00</created_at>
    <id type="integer">152</id>
    <label>CDN Edge group name</label>
    <updated_at type="datetime">2012-04-18T11:33:01+00:00</updated_at>
    <assigned_locations type="array">
      <location>
        <id type="integer">146</id>
        <price type="float">0.3</price>
        <region>DC</region>
        <city>Washington</city>
        <latitude type="float">38.895</latitude>
        <country>US</country>
        <deleted type="boolean">false</deleted>
        <longitude type="float">-77.0367</longitude>
      </location>
    </assigned_locations>
  </edge_group>
...<location><assigned_locations><available_locations type="array"><location><description>abc</description><id type="integer">2</id><price type="float">10.0</price><region>T2</region><city>bangor</city><latitude type="float">54.65</latitude><country>GB</country><deleted type="boolean">false</deleted><longitude type="float">-5.7</longitude></location><location><location>
...</navailable_locations></edge_group>

Explanation of the data returned:
available_locations - an array of all available locations
assigned_locations - an array of locations, which are assigned to the group
city - city where the edge server is located
region - region where the edge server is located
price - price per GB of sold excess bandwidth
latitude - latitude of the server location
longitude - longitude of the server location
country - country codes related to country_access_policy in ISO 3166-1 alpha-2 format
updated_at - date when the location was updated
deleted - true if the location is deleted; otherwise false
id - the ID of location in the OnApp CP data base
created_at- date, when the location was created
description - optional description of the location

23.4 Add CDN Edge Group

To create an edge group, use the following API call:
POST /edge_groups.xml
POST /edge_groups.json
XML Request example
curl -i -X POST -u user:userpass http://onapp.test/edge_groups.xml
-d '<edge_group><label>az_3</label></edge_group>' -H
'Accept:application/xml' -H 'Content-type:application/xml'

**JSON Request example**
curl -i -X POST -u user:userpass http://onapp.test/edge_groups.json
-d '{"edge_group":{"label":"az_4"}}' -H 'Accept:application/json' -H 'Content-type:application/json'

**Parameters:**

*label* * - the name of new group

## 23.5 Edit CDN Edge Group

You can edit the *label* of the edge group:

PUT /edge_groups/:id.xml
PUT /edge_groups/:id.json

**XML request example**
curl -i -X PUT -u user:userpass
http://onapp.test/edge_groups/:id.xml -d
'<edge_group><label>az_5</label></edge_group>' -H 'Accept:application/xml' -H 'Content-type:application/xml'

**JSON Request example**

curl -i -X PUT -u user:userpass
http://onapp.test/edge_groups/:id.json -d
'{"edge_group":{"label":"az_6"}}' -H 'Accept:application/json' -H 'Content-type:application/json'

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

## 23.6 Delete CDN Edge Group

To delete the edge group, use the following request:

DELETE /edge_groups/:id.xml
DELETE /edge_groups/:id.json

**XML Request example**
curl -i -X DELETE -u user:userpass
http://onapp.test/edge_groups/:id.xml -H 'Accept:application/xml' -H 'Content-type:application/xml'
JSON Request example

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

Where you have to specify ID of a CDN group you want to delete in the URL.

Be careful when deleting an edge group which is associated with CDN resources.

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

23.7 Assign Location to CDN Edge Group

Check the ID of the required location and assign it to the group with the following API call:

POST /edge_groups/:edge_group_id/assign.xml
POST /edge_groups/:edge_group_id/assign.json

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_groups/1/assign.xml -d
'<location>175</location>' -H 'Accept:application/xml' -H 'Content-type:application/xml'

JSON Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_groups/1/assign.json -d '{"location":"175"}'
-H 'Accept:application/json' -H 'Content-type:application/json'

Where:
location * - input the ID of the required location
You can retrieve the list of location IDs with the Get CDN Edge Group Details API call.

23.8 Unassign Location From CDN Edge Group

To remove a location from the group, use the following method:

POST /edge_groups/:edge_group_id/unassign.xml
POST /edge_groups/:edge_group_id/unassign.json

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_groups/1/unassign.xml -d
23.9 Modify CDN Edge Group

To modify CDN edge group, run:

POST /edge_groups/:edge_group_id/modify.xml
POST /edge_groups/:edge_group_id/modify.json

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_groups/1/modify.xml -d
"<locations type="array">"<location>123</location><location>123</location><location>123</location></locations>" -H 'Accept:application/xml' -H 'Content-type:application/xml'

JSON Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_groups/1/modify.json -d
"{"locations":["123","456","789"]}" -H 'Accept:application/json' -H 'Content-type:application/json'

Where you have to specify IDs of locations you want to see eventually in the required CDN resource (at least one ID is required).
You can retrieve the list of location IDs with the Get CDN Edge Group Details API call.

23.10 Search CDN Edge Groups

To search for a specific CDN Edge Group, use the following request:

GET /edge_groups.xml?q=label
GET /edge_groups.json?q=label

Where you have to specify the Edge Group label.

XML Request example
**WHITELISTED IPS**

**EDIT WHITELISTED IP**

```bash
```

**JSON Request example**

```bash
```

The request will search for the Edge Group with the test label.

**XML output example:**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<edge_groups type="array">
  <edge_group>
    <created_at type="datetime">2013-08-12T11:39:09+03:00</created_at>
    <id type="integer">228</id>
    <label>PItest</label>
    <updated_at type="datetime">2013-08-12T11:39:09+03:00</updated_at>
    <cdn_reference type="integer">426796953</cdn_reference>
  </edge_group>
  <edge_group>
    <created_at type="datetime">2013-08-19T14:32:54+03:00</created_at>
    <id type="integer">232</id>
    <label>TestPI</label>
    <updated_at type="datetime">2013-08-19T14:32:54+03:00</updated_at>
    <cdn_reference type="integer">668633450</cdn_reference>
  </edge_group>
</edge_groups>
```

Where:
- `created_at` - the date when the edge group was created
- `id` - the resource ID in the database
- `label` - the edge group label
- `updated_at` - the date when the edge group was updated
- `cdn_reference` - the identifier in database
24 CDN EDGE SERVERS

CDN edge servers are the virtual server which form a Content Delivery Network. In this network the web content is cached and delivered to end users from the server which is closest to the user or has the best availability.

24.1 Get List of CDN Edge Servers

To view all edge servers in the cloud with their details, use the following request:

GET /edge_servers.xml
GET /edge_servers.json

XML Request example:
```
curl -i -X GET -u user:userpass http://onapp.test/edge_servers.xml
```

JSON Request example:
```
curl -i -X GET -u user:userpass http://onapp.test/edge_servers.json
```

To get the list of HTTP edge servers:

XML Request example:
```
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.xml?type=http
```

JSON Request example:
```
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.json?type=http
```

To get the list of streaming edge servers:

XML Request example:
```
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.xml?type=streaming
```

JSON Request example:
```
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.json?type=streaming
```

XML Output example
```
<edge_servers type="array">
<edge_server>
<add_to_marketplace type="boolean">true</add_to_marketplace>
<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">true</built>
<cpu_shares type="integer">7</cpu_shares>
<cpu_sockets nil="true"/>
<cpu_threads nil="true"/>
<cpu_units type="integer">140</cpu_units>
<cpus type="integer">1</cpus>
<created_at type="datetime">2015-02-10T14:19:45+02:00</created_at>
<customer_network_id nil="true"/>
<deleted_at nil="true"/>
<edge_server_type>streaming</edge_server_type>
<enable_autoscale nil="true"/>
<enable_monitis type="boolean">false</enable_monitis>
<firewall_notrack type="boolean">true</firewall_notrack>
<hypervisor_id type="integer">25</hypervisor_id>
<id type="integer">3781</id>
<identifier>kcs046otoxbr0</identifier>
<initial_root_password>3yV4or11B1Le</initial_root_password>
<initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
<label>qawsedrf</label>
<local_remote_access_ip_address>109.123.91.36</local_remote_access_ip_address>
<local_remote_access_port nil="true"/>
<locked type="boolean">false</locked>
<memory type="integer">2054</memory>
<min_disk_size type="integer">20</min_disk_size>
<note nil="true"/>
<operating_system>linux</operating_system>
<operating_system_distro>ubuntu</operating_system_distro>
<pREFERRED_HVS_TYPE="array"/>
<recovery_mode type="boolean">false</recovery_mode>
<remote_access_password>GPRdQyq28jVR</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">11</template_id>
<template_label>debian-6.0-x64-1.14-xen.kvm.kvm_virtio.tar.gz</template_label>
<updated_at type="datetime">2015-03-05T10:18:53+02:00</updated_at>
<user_id type="integer">4</user_id>
<vip nil="true"/>
<xen_id nil="true"/>
<ip_addresses type="array">
<ip_address>
<address>109.123.91.154</address>
</ip_address>
</ip_addresses>
OnApp 5.3 API Guide  v2

WHITELIST IPS

<broadcast>109.123.91.191</broadcast>
<created_at type="datetime">2014-01-14T14:19:52+02:00</created_at>
<customer_network_id nil="true"/>
<disallowed_primary type="boolean">false</disallowed_primary>
<gateway>109.123.91.129</gateway>
<hypervisor_id nil="true"/>
?id type="integer">25</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">2014-11-06T17:10:35+02:00</updated_at>
<user_id nil="true"/>
<free type="boolean">false</free>
<netmask>255.255.255.192</netmask>
</ip_address>
</ip_addresses>

Explanation of the data returned:
add_to_marketplace - true if this edge server is added to the marketplace; otherwise false
admin_note - an optional reminder for this VS created by an administrator
allow_resize_without_reboot - true if adjusting resource allocation without reboot is possible; 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 server is booted; otherwise false
cpu_shares - the CPU priority percentage
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.
cpus - number of CPU cores allocated to this edge server
**created_at** - the date when the CDN edge server was created in the [YYYY][MM][DD]T[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** - false; not available for edge servers

**enable_monitis** - deprecated attribute; will be removed in upcoming release

**firewall_notrack** - true if the NOTRACK rule is set in iptables

**hypervisor_id** - the ID of the compute resource, on which the server is deployed

**id** - the edge server ID in OnApp CP database

**identifier** - the edge server identifier

**initial_root_password** - the server root password

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

**label** - an arbitrary name of the edge server

**local_remote_access_port** - the port ID used for console access

**locked** - true if locked; otherwise false

**memory** - the amount of RAM resources allocated to this edge server

**min_disk_size** - minimum disk space required by the template

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

**operating_system** - type of operating system

**operating_system_distro** - the distribution of the operating system

**preferred_hvs** - the array of preferable compute resources based on compute zone that meet some VS configuration settings

**recovery_mode** - true if the server is booted in the recovery mode; otherwise false

**remote_access_password** - the password for remote access

**service_password** - service account password

**state** - deprecated attribute; will be removed in upcoming release

**storage_server_type** - true if this is a storage server

**strict_virtual_machine_id** - the ID of a virtual server (or edge server) that will never reside on the same compute resource with this server

**suspended** - true if suspended; otherwise false

**template_id** - the ID of the template, on which the edge server is based

**template_label** - label of the template on which the server is based; currently - OnApp CDN compute resource

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

**user_id** - the ID of the user, who is the server owner

**vip** - true if the server has VIP status for migration; otherwise false

**xen_id** - the edge server ID set by the virtualization engine

**ip_addresses** - an array of assigned IP addresses with their details assigned to this edge 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]T[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
WHITELIST IPS

- **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
- **monthly_bandwidth_used** - VS monthly bandwidth in KB
- **total_disk_size** - total disk space in GB of primary and swap disks
- **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**
- **edge_status** - the edge server status
- **cdn_reference** - the identifier in database

### 24.2 Get CDN Edge Server Details

To view the edge server details:

GET /edge_servers/:id.xml
GET /edge_servers/:id.json

**XML Request example:**
```
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers/:id.xml
```

**JSON Request example:**
```
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers/:id.json
```

To get the list of HTTP edge servers

**XML Request example:**
```
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.xml?type=http
```

**JSON Request example:**
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.json?type=http

To get the list of streaming edge servers:

XML Request example:
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.xml?type=streaming

JSON Request example:
curl -i -X GET -u user:userpass
http://onapp.test/edge_servers.json?type=streaming

XML Output example
<edge_servers type="array">
  <edge_server>
    <add_to_marketplace type="boolean">true</add_to_marketplace>
    <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">true</built>
    <cpu_shares type="integer">7</cpu_shares>
    <cpusockets nil="true"/>
    <cpu_threads nil="true"/>
    <cpu_units type="integer">140</cpu_units>
    <cpus type="integer">1</cpus>
    <created_at type="datetime">2015-02-10T14:19:45+02:00</created_at>
    <customer_network_id nil="true"/>
    <deleted_at nil="true"/>
    <edge_server_type>streaming</edge_server_type>
    <enable_autoscale nil="true"/>
    <enable_monitis type="boolean">false</enable_monitis>
    <firewall_notrack type="boolean">true</firewall_notrack>
    <hypervisor_id type="integer">25</hypervisor_id>
    <id type="integer">3781</id>
    <identifier>kcs046otoxbr0</identifier>
    <initial_root_password>3yV4Or1B1Le</initial_root_password>
    <initial_root_password_encrypted type="boolean">false</initial_root_password_encrypted>
    <label>qawsedrf</label>
    <local_remote_access_ip_address>109.123.91.36</local_remote_access_ip_address>
    <local_remote_access_port nil="true"/>
    <locked type="boolean">false</locked>
    <memory type="integer">2054</memory>
    <min_disk_size type="integer">20</min_disk_size>
    <note nil="true"/>
    <operating_system>linux</operating_system>
    <operating_system_distro>ubuntu</operating_system_distro>
<preferred_hvs type="array"/>
<recovery_mode type="boolean">false</recovery_mode>
<remote_access_password>GPRdQyq28jVR</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">11</template_id>
<template_label>debian-6.0-x64-1.14-xen.kvm.kvm_virtio.tar.gz</template_label>
<updated_at type="datetime">2015-03-05T10:18:53+02:00</updated_at>
<user_id type="integer">4</user_id>
<vip nil="true"/>
<xen_id nil="true"/>
<ip_addresses type="array">
  <ip_address>
    <address>109.123.91.154</address>
    <broadcast>109.123.91.191</broadcast>
    <created_at type="datetime">2014-01-14T19:52+02:00</created_at>
    <customer_network_id nil="true"/>
    <disallowed_primary type="boolean">false</disallowed_primary>
    <gateway>109.123.91.129</gateway>
    <hypervisor_id nil="true"/>
    <id type="integer">25</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">2014-11-06T17:10:35+02:00</updated_at>
    <user_id nil="true"/>
    <free type="boolean">false</free>
    <netmask>255.255.255.192</netmask>
  </ip_address>
</ip_addresses>
<monthly_bandwidth_used>0</monthly_bandwidth_used>
<total_disk_size type="integer">20</total_disk_size>
<price_per_hour type="float">20540.0</price_per_hour>
<price_per_hour_powered_off type="float">0.0</price_per_hour_powered_off>
<support_incremental_backups type="boolean">true</support_incremental_backups>
<cpu_priority type="integer">7</cpu_priority>
<edge_status>Active</edge_status>
<cdn_reference type="integer">276964394</cdn_reference>
</edge_server>
Explanation of the data returned:

- **add_to_marketplace** - true if this edge server is added to the marketplace; otherwise false
- **admin_note** - an optional reminder for this VS created by an administrator
- **allow_resize_without_reboot** - true if adjusting resource allocation without reboot is possible; 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 server is booted; otherwise false
- **cpu_shares** - the CPU priority percentage
- **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.
- **cpus** - number of CPU cores allocated to this edge server
- **created_at** - the date when the CDN edge server was created in the [YYYY][MM][DD][T][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; not available for edge servers
- **enable_monitis** - deprecated attribute; will be removed in upcoming release
- **firewall_notrack** - true if the NOTRACK rule is set in iptables
- **hypervisor_id** - the ID of the compute resource, on which the server is deployed
- **id** - the edge server ID in OnApp CP database
- **identifier** - the edge server identifier
- **initial_root_password** - the server root password
- **initial_root_password_encrypted** - true, if the server root password is encrypted, otherwise false
- **label** - an arbitrary name of the edge server
- **local_remote_access_port** - the port ID used for console access
- **locked** - true if locked; otherwise false
- **memory** - the amount of RAM resources allocated to this edge server
- **min_disk_size** - minimum disk space required by the template
- **note** - an optional reminder for this VS made by a user account
- **operating_system** - type of operating system
- **operating_system_distro** - the distribution of the operating system
- **preferred_hvs** - the array of preferable compute resources based on compute zone that meet some VS configuration settings
- **recovery_mode** - true if the server is booted in the recovery mode; otherwise false
- **remote_access_password** - the password for remote access
- **service_password** - service account password
- **state** - deprecated attribute; will be removed in upcoming release
- **storage_server_type** - true if this is a storage server
- **strict_virtual_machine_id** - the ID of a virtual server (or edge server) that will never reside on the same compute resource with this server
suspended - true if suspended; otherwise false

template_id - the ID of the template, on which the edge server is based

template_label - label of the template on which the server is based; currently -

OnApp CDN compute resource

updated_at - the date when the CDN edge server was updated in the

[YYYY][MM][DD]T[hh][mm][ss]Z format

user_id - the ID of the user, who is the server owner

vip - true if the server has VIP status for migration; otherwise false

xen_id - the edge server ID set by the virtualization engine

ip_addresses - an array of assigned IP addresses with their details assigned to

this edge 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]T[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]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 - total disk space in GB of primary and swap disks

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

deployment_status - the edge server status

cdn_reference - the identifier in database
# 24.3 Add CDN Edge Server

To create an edge server, use the following API call:

**POST /edge_servers.xml**

**POST /edge_servers.json**

**XML Request example**

```bash
curl -i -X POST -d
'<edge_server><label>az_CDN_test</label><add_to_marketplace>true</add_to_marketplace><cpus>1</cpus><data_store_group_primary_id>2</data_store_group_primary_id><primary_network_group_id>3</primary_network_group_id><cpu_shares>1</cpu_shares><memory>2048</memory><required_virtual_machine_build>1</required_virtual_machine_build><hypervisor_group_id>1</hypervisor_group_id><hypervisor_id>1</hypervisor_id><required_ip_address_assignment>1</required_ip_address_assignment><primary_disk_size>20</primary_disk_size><rate_limit>0</rate_limit><cdn_location>5</cdn_location><edge_server_type>http</edge_server_type></edge_server>'

-u user:userpass

http://onapp.test/edge_servers.xml

-H 'Accept: application/xml'

-H 'Content-type: application/xml'
```

**JSON Request example**

```bash
curl -i -X POST -d
'{"edge_server":{"label":"az_CDN_test","add_to_marketplace":"true","cpus":"1","data_store_group_primary_id":"2","primary_network_group_id":"3","cpu_shares":"1","memory":"2048","required_virtual_machine_build":"1","hypervisor_group_id":"1","hypervisor_id":"1","required_ip_address_assignment":"1","primary_disk_size":"20","rate_limit":"0","cdn_location":"5","edge_server_type":"http"}}'

-u user:userpass

http://onapp.test/edge_servers.json

-H 'Accept: application/json'

-H 'Content-type: application/json'
```

**Where:**

- **label** - a unique name of your CDN edge server. The label can consist of letters [A-Za-z], digits [0-9], dash [-], lower dash [ _ ], space character [ ], at sign [@], round brackets [()], slashes [/], comma [,] and dot [.]. You can use both lower- and uppercase letters. The label should begin with an alphanumeric character or lower dash [ _ ]
- **hypervisor_id** - indicate the ID of the compute resource, on which the server will be deployed
- **hypervisor_group_id** - indicate the compute zone ID
- **cpus** - the amount of CPU cores allocated to this edge server
- **cpu_shares** - the percentage of allocated CPU priority resource
- **memory** - the amount of RAM, which you want to allocate to this edge server
- **primary_disk_size** - the size in GB of the primary disk
- **data_store_group_primary_id** - specify the ID of a data store zone, where you want to locate the disk of your server. If not specified – the system will select the data store zone with higher available capacity
OnApp 5.3 API Guide

WHITELIST IPS

### primary_network_group_id
- indicate the network zone ID

### required_virtual_machine_build
- set "1" to build the server automatically after creation. Otherwise set "0"

### required_ip_address_assignment
- set "1" if you want IP address to be assigned automatically after creation. Otherwise set "0"

### add_to_marketplace
- set "true", if the edge server is added to marketplace; otherwise set "false". The default value is "false".

### cdn_location
- the ID of the CDN location. Use the following API call to find the ID 'Get List of CDNLocations for Location Group'. The parameter is optional. If not set, the edge server will be assigned to the first CDN Location in its Location Group.

### edge_server_type
- set http or streaming server type

**ATTENTION!** Creating a Streaming Edge or Storage server will result in an additional monthly charge. You will be charged 50$ per month for deploying this Streaming server once it is provisioned.

#### 24.3.1 Page History

**v. 3.3.1**

- cdn_location parameter
- edge_server_type

#### 24.4 Edit CDN Edge Server

To change the server label and resource allocation:

PUT /edge_servers/:id.xml
PUT /edge_servers/:id.json

**XML Request example**

curl -i -X PUT -d
  
  '<edge_server><label>az_CDN_test_1</label><add_to_marketplace>true</add_to_marketplace><cpus>1</cpus><cpu_shares>10</cpu_shares><memory>512</memory></edge_server>'

http://onapp.test/edge_servers/:id.xml

-H 'Accept: application/xml'
-H 'Content-type: application/xml'

**JSON Request example**

curl -i -X PUT -d

'{"edge_server":{"label":"az_CDN_test_3","add_to_marketplace":"true","cpus":"1","cpu_shares":"20","memory":"512"}}'

http://onapp.test/edge_servers/:id.json

-H 'Accept: application/json'
-H 'Content-type: application/json'

**Where:**
**24.5 Reboot CDN Edge Server**

To reboot the edge server:

- POST /edge_servers/:edge_server_id/reboot.xml
- POST /edge_servers/:edge_server_id/reboot.json

**XML Request example**
```
curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/reboot.xml -H
'Accept: application/xml' -H 'Content-type: application/xml'
```

**JSON Request example**
```
curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/reboot.json -H
'Accept: application/json' -H 'Content-type: application/json'
```

**24.6 Reboot CDN Edge Server in Recovery**

To reboot the edge server in recovery mode with a temporary login ("root") and password ("recovery"), use the following API calls:

- POST /edge_servers/:edge_server_id/reboot.xml
- POST /edge_servers/:edge_server_id/reboot.json

**XML Request example**
```
*curl -i -X POST -u user:userpass '<mode>recovery</mode>' --url
http://onapp.test/edge_servers/:edge_server_id/reboot.xml
```

**JSON Request example**
```
*curl -i -X POST -u user:userpass '{"mode":"recovery"}' --url
http://onapp.test/edge_servers/:edge_server_id/reboot.json
```
24.7 Start up CDN Edge Server

POST /edge_servers/:edge_server_id/startup.xml
POST /edge_servers/:edge_server_id/startup.json

XML Request example
  curl -i -X POST -u user:userpass
  http://onapp.test/edge_servers/:edge_server_id/startup.xml -H
  'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example
  curl -i -X POST -u user:userpass
  http://onapp.test/edge_servers/:edge_server_id/startup.json -H
  'Accept: application/json' -H 'Content-type: application/json'

24.8 Shut down CDN Edge Server

To terminate the edge server gracefully:
POST /edge_servers/:edge_server_id/shutdown.xml
POST /edge_servers/:edge_server_id/shutdown.json

XML Request example
  curl -i -X POST -u user:userpass
  http://onapp.test/edge_servers/:edge_server_id/shutdown.xml -H
  'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example
  curl -i -X POST -u user:userpass
  http://onapp.test/edge_servers/:edge_server_id/shutdown.json -H
  'Accept: application/json' -H 'Content-type: application/json'

24.9 Stop CDN Edge Server

To terminate the edge server forcefully:
POST /edge_servers/:edge_server_id/stop.xml
POST /edge_servers/:edge_server_id/stop.json

XML Request example
  curl -i -X POST -u user:userpass
  http://onapp.test/edge_servers/:edge_server_id/stop.xml -H 'Accept:
  application/xml' -H 'Content-type: application/xml'

JSON Request example
  curl -i -X POST -u user:userpass
24.10 Rebuild CDN Edge Server

To rebuild (or build manually) the edge server, use the following request:

**POST /edge_servers/:edge_server_id/build.xml**

**POST /edge_servers/:edge_server_id/build.json**

**XML Request example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/build.xml
-d '<?xml version="1.0" encoding="UTF-8"?><edge_server><template_id>398</template_id><required_startup>1</required_startup></edge_server>'
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

**JSON Request example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/build.json
-d '{"edge_server":{"template_id":"398","required_startup":"1"}}'
-H 'Accept: application/json'
-H 'Content-type: application/json'
```

Where:
- **template_id** - the ID of the template on which this server will be based
- **required_startup** - set "1" to start up the server automatically after build, otherwise set "0"

24.11 Suspend CDN Edge Server

**POST /edge_servers/:edge_server_id/suspend.xml**

**POST /edge_servers/:edge_server_id/suspend.json**

**XML Request example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/suspend.xml
-H 'Accept: application/xml'
-H 'Content-type: application/xml'
```

**JSON Request example**

```bash
curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/suspend.json
-H 'Accept: application/json'
-H 'Content-type: application/json'
```

To unsuspend the server, run the request again.
24.12 Rerun CDN Edge Server Creation Scripts

When an edge server is built, the system will run the scripts for creation of an edge server. You can do it manually, using the following request:

POST /edge_servers/:edge_server_id/rerun_edge_scripts.xml
POST /edge_servers/:edge_server_id/rerun_edge_scripts.json

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/rerun_edge_scripts.xml
-H 'Accept:application/xml' -H 'Content-type:application/xml'

JSON Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/rerun_edge_scripts.json
-H 'Accept:application/json' -H 'Content-type:application/json'

24.13 Unlock CDN Edge Server

To unlock the edge server:

POST /edge_servers/:edge_server_id/unlock.xml
POST /edge_servers/:edge_server_id/unlock.json

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/unlock.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example

curl -i -X POST -u user:userpass
http://onapp.test/edge_servers/:edge_server_id/unlock.json
-H 'Accept: application/json' -H 'Content-type: application/json'

24.14 Delete CDN Edge Server

DELETE /edge_servers/:id.xml
DELETE /edge_servers/:id.json

XML Request example
curl -i -X DELETE -u user:userpass

**JSON Request example**

curl -i -X DELETE -u user:userpass
http://onapp.test/edge_servers/:edge_server_id.json -H 'Content-type: application/json'

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

### 24.15 Migrate CDN Edge Server

To migrate an edge server to another compute resource, use the following request:

**POST /edge_servers/:edge_server_id/migrate.xml**

**POST /edge_servers/:edge_server_id/migrate.json**

**XML Request example**

```bash
*curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d
"<edge_server><destination>1</destination><cold_migrate_on_rollback>1</cold_migrate_on_rollback></edge_server>" --url
http://onapp.test/edge_servers/:edge_server_id/migrate.xml*
```

**JSON Request example**

```bash
*curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d
'{"edge_server":{"destination":"1","cold_migrate_on_rollback":"1"}}' --url
http://onapp.test/edge_servers/:edge_server_id/migrate.json*
```

Where:

- **destination** - the ID of a target compute resource, to which you migrate the edge server
- **cold_migrate_on_rollback** - set 1 if you wish to switch to a cold migration if hot migration fails, otherwise set 0.

### 24.16 Segregate CDN Edge Server

To segregate an edge server (that is, instruct it never to reside on the same compute resource as another VS or edge server), use the following method:

**POST /edge_servers/:edge_server_id/strict_vm.xml**

**POST /edge_servers/:edge_server_id/strict_vm.json**
**WHITELIST IPS**

**EDIT WHITELISTED IP**

**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"?>
<virtual_machine><strict_virtual_machine_id>bb6oa3eqdzpcgl</strict_virtual_machine_id></virtual_machine>' --url http://onapp.test/edge_servers/:edge_server_id/strict_vm.xml* 
```

**JSON Request example**

```bash
*curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"virtual_machine":{"strict_virtual_machine_id":"gv03xz1x31t53h"}}' --url http://onapp.test/edge_servers/:edge_server_id/strict_vm.json* 
```

Where:

- `strict_virtual_machine_id` - the ID of virtual server you wish to segregate from the given edge server

### 24.17 Open CDN Edge Server Console

To open an edge server console:

1. Run the following request:
   ```bash
   GET /edge_servers/:edge_server_id/console.xml
   GET /edge_servers/:edge_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:
   ```bash
   *http://onapp.test/console_remote/\[remote_key_parameter_value\]*
   ```

### 24.18 Change CDN Edge Server Owner

Use the following request to reassign an edge server to another user:

```bash
POST /edge_servers/:edge_server_id/change_owner.xml
POST /edge_servers/:edge_server_id/change_owner.json
```

**XML Request example**

```bash
*curl -i -X POST -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass -d '<user_id>4</user_id>' --url http://onapp.test/edge_servers/:edge_server_id/change_owner.xml* 
```

**JSON Request example**

```bash
*curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"user_id":4}' --url http://onapp.test/edge_servers/:edge_server_id/change_owner.json* 
```
application/json' -u user:userpass -d '{'user_id':'1'}" --url
http://onapp.test/edge_servers/:edge_server_id/change_owner.json*

Where:
user_id * – input ID of a new server owner

24.19 Set VIP Status for CDN Edge Server

To give your edge server a migration priority, set the VIP status for it with the following request:
POST /edge_servers/:edge_server_id/set_vip.xml
POST /edge_servers/:edge_server_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.

24.20 Add/Edit Admin/User Note for CDN Edge Server

To edit/make an admin note, use the following request:
PUT /edge_servers/:edge_server_id.xml
PUT /edge_servers/:edge_server_id.json
XML Request example
  curl -i -X PUT -u user:userpass
http://onapp.test/edge_servers/:edge_server_id.xml -d
  '<edge_server><admin_note>agfagwe tiuuytjgh yuytu</admin_note></edge_server>' -H 'Accept:application/xml' -H 'Content-type:application/xml'

JSON Request example
  curl -i -X PUT -u user:userpass
http://onapp.test/edge_servers/:edge_server_id.json -d '{"edge_server":{"admin_note":"kjfjhtrtjtt"}}' -H 'Accept:application/json' -H 'Content-type:application/json'

Where:
admin_note - enter the text of your note.

Add/Edit User Note XML Request example
curl -i -X PUT -u user:userpass
http://onapp.test/edge_servers/:edge_server_id.xml -d
'<?xml version="1.0" encoding="utf-8"?>
<edge_server>
<note>changed</note>
</edge_server>' -H
'Accept:application/xml' -H 'Content-type:application/xml'

Add/Edit User Note JSON Request example
curl -i -X PUT -u user:userpass
http://onapp.test/edge_servers/:edge_server_id.json -d
'{"edge_server":{"note":"kjfjhtrtjtt"}}' -H
'Accept:application/json' -H 'Content-type:application/json'

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

24.21 CDN Edge Server Disks

Since CDN edge servers are virtual servers in their essence, you may perform all the same actions with edge servers' disks as with VS disks, except POST and DELETE. The only difference would be in the routes for the following requests:

To view the edge server disks:
GET /edge_servers/:edge_server_id/disks.xml
GET /edge_servers/:edge_server_id/disks.json
Parameters description and output example.
For other possible requests, refer to corresponding sections of Disks chapter.

24.22 CDN Edge Server Network Interfaces

Here is the list of API calls for managing CDN edge servers' network interfaces. Edge servers' network interfaces have the same attributes as network interfaces of virtual servers.

To get the list of network interfaces allocated to this particular edge server:
GET /edge_servers/:edge_server_id/network_interfaces.xml
GET /edge_servers/:edge_server_id/network_interfaces.json
To get a particular network interface details:
GET /edge_servers/:edge_server_id/network_interfaces/:id.xml
GET /edge_servers/:edge_server_id/network_interfaces/:id.json
To edit network interface details:
PUT /edge_servers/:edge_server_id/network_interfaces/:id.xml
PUT /edge_servers/:edge_server_id/network_interfaces/:id.json
To add a new network interface:
POST /edge_servers/:edge_server_id/network_interfaces.xml
POST /edge_servers/:edge_server_id/network_interfaces.json
To delete a network interface from the edge server:
DELETE /edge_servers/:edge_server_id/network_interfaces/:id.xml
DELETE /edge_servers/:edge_server_id/network_interfaces/:id.json

XML Output example
<?xml version="1.0" encoding="UTF-8"?>

<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>pdfjrtpkday9e1</identifier>
</network_interface>

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
24.23 CDN Edge Server IP Address Joins

An IP address allocated to an edge server is an IP address join. Use the following methods to view, assign and delete IP address joins of your CDN edge servers.

To get the list of IP address assignments for a particular edge server:
GET /edge_servers/:edge_server_id/ip_addresses.xml
GET /edge_servers/:edge_server_id/ip_addresses.json

XML Request Example
```
```

Json Request Example
```
```

To assign an IP Address to an edge server:
POST /edge_servers/:edge_server_id/ip_addresses.xml
POST /edge_servers/:edge_server_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_join><ip_address_id>7</ip_address_id><network_interface_id>131</network_interface_id></ip_address_join>' --url http://onapp.test/edge_servers/:edge_server_id/ip_addresses.xml
```

Json Request Example
```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass -d '{"ip_address_join":{"ip_address_id":"7", "network_interface_id":"131"}}' --url http://onapp.test/edge_servers/:edge_server_id/ip_addresses.json
```

XML Response Example
```
<?xml version="1.0" encoding="UTF-8"?>
<ip_address_join>
  <created_at type="datetime">2013-10-31T13:04:05+03:00</created_at>
  <id type="integer">173</id>
</ip_address_join>
```
<ip_address_id type="integer">7</ip_address_id>
<network_interface_id type="integer">131</network_interface_id>
<updated_at type="datetime">2013-10-31T13:04:05+03:00</updated_at>

<ip_address>
    <address>1.1.1.3</address>
    <broadcast>1.1.1.255</broadcast>
    <created_at type="datetime">2013-08-07T13:29:09+03:00</created_at>
    <customer_network_id nil="true"/>
    <disallowed_primary type="boolean">false</disallowed_primary>
    <gateway>1.1.1.1</gateway>
    <hypervisor_id nil="true"/>
    <id type="integer">7</id>
    <ip_address_pool_id nil="true"/>
    <network_address>1.1.1.0</network_address>
    <network_id type="integer">1</network_id>
    <pxe type="boolean">false</pxe>
    <updated_at type="datetime">2013-08-07T13:29:09+03:00</updated_at>
    <user_id nil="true"/>
    <free type="boolean">false</free>
    <netmask>255.255.255.0</netmask>
</ip_address>
</ip_address_join>

Where:
created_at - the date when the record was created in DB
id - the IP address join ID
ip_address_id - the IP address ID
network_interface_id - the network interface ID
updated_at - the date when the record was updated in DB
ip_address - the array of IP address details
address - the IP address
broadcast - a logical address at which all devices connected to a multiple-access communications network are enabled to receive datagrams.
customer_network_id - the ID of the customer network
disallowed_primary - true if this address is not set as primary (for VS build), otherwise false
gateway - gateway address
hypervisor_id - the ID of the compute resource
ip_address_pool_id - the ID of the IP address poll to which this join belongs
network_address - the address of a VLAN network address that will be associated with this IP address pool
network_id - the ID of the network
pxe - true, if this address can be used for cloudbooting a compute resource
free - true if free, otherwise false
netmask - netmask for the IP address

To delete an IP address assignment from a particular edge server:
DELETE /edge_servers/:edge_server_id/ip_addresses/:id.xml
DELETE /edge_servers/:edge_server_id/ip_addresses/:id.json

XML Request Example
```
curl -i -X DELETE -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url http://onapp.test/edge_servers/: edge_server_id/ip_addresses/: ip_address_join_id.xml
```

Json Request Example
```
curl -i -X DELETE -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/edge_servers/: edge_server_id/ip_addresses/: ip_address_join_id.json
```

Where:
data_store_id - the ID of the data store, which is attached to the compute resource
hypervisor_id - reserved parameter
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

## 24.24 Rebuild Network for CDN Edge Server

It is required to rebuild network after any changes on IP address joins or network interfaces. To rebuild network, use the following request:

POST /edge_servers/:edge_server_id/rebuild_network.xml
POST /edge_servers/:edge_server_id/rebuild_network.json

XML Request example
```
```

JSON Request example
```
curl -X POST -u user:userpass --url 'http://onapp.test/edge_servers/:edge_server_id/rebuild_network.json?force=1&shutdown_type=hard&required_startup=1' -H 'Accept:
```
Where:

- **storage_server_id** - ID of the edge server
- **shutdown_type** - type of the edge server shutdown: hard, graceful or soft
- **required_startup** - set 1 to start up the server automatically after build, otherwise set 0

### 24.25 Get CDN Edge Server Billing Statistics

You can view the billing statistics for a particular edge server using the following request:

GET `/edge_servers/:edge_server_id/vm_stats.xml`  
GET `/edge_servers/:edge_server_id/vm_stats.json`

Define a shorter period by setting Start and End time in the API call:  

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<vm_stats type="array">
    <vm_hourly_stat>
        <created_at type="datetime">2011-11-01T00:00:13Z</created_at>
        <updated_at type="datetime">2011-11-01T00:00:13Z</updated_at>
        <stat_time type="datetime">2011-11-01T00:00:00Z</stat_time>
        <total_cost type="float">0.0</total_cost>
        <id type="integer">9582</id>
        <vm_billing_stat_id type="integer">7795</vm_billing_stat_id>
        <user_id type="integer">1</user_id>
        <billing_stats>
            <virtual_machines type="array">
                <virtual_machine>
                    <label>QAVP XEN serveridze</label>
                    <costs type="array">
                        <cost>
                            <value type="integer">100</value>
                        </cost>
                    </costs>
                </virtual_machine>
            </virtual_machines>
        </billing_stats>
    </vm_hourly_stat>
</vm_stats>
```
<resource_name>cpu_shares</resource_name>
<cost type="float">0.0</cost>
</cost>
...
<cost></cost>
...
</costs>
<id type="integer">237</id>
</virtual_machine>
</virtual_machines>
<network_interfaces type="array">
<network_interface>
<label>eth0</label>
<costs type="array">
<cost>
<value type="integer">1</value>
</cost>
...
<cost></cost>
...
</costs>
<id type="integer">254</id>
</network_interface>
</network_interfaces>
<disks type="array">
<disk>
<label>#499</label>
<costs type="array">
<cost>
<value type="integer">20</value>
</cost>
...
<cost></cost>
...
</costs>
<id type="integer">499</id>
</disk>
</disks>
<edge_servers type="array">
<edge_server>
<label>QA VP XEN serveridze</label>
<costs type="array">
<cost>
<value type="integer">14</value>
</cost>
<resource_name>template</resource_name>
<cost type="float">0.0</cost>
</cost>
</costs>
{id type="integer">237</id>
</edge_server>
</billing_stats>
<usage_cost type="float">0.0</usage_cost>
</virtual_machine_id type="integer">237</virtual_machine_id>
<currency_code>USD</currency_code>
<vm_resources_cost type="float">0.0</vm_resources_cost>
</vm_hourly_stat>
</vm_stats>

Where:
created_at – the timestamp in DB when this record was created
updated_at – the date when these statistics were updated
cost – the total amount of money owed by this particular edge server for the
resources spent at stat_time
stat_time – the particular hour for which these statistics were generated
id - the ID of these statistics
user_id - the ID of edge server owner
currency_code - currency in which this virtual machine is charged within the
billing plan
billing_stats - an array of billing details for the resources used by this edge
server
virtual_machine - an array of edge server billing details:
label – name of the edge server
costs- an array of edge server 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 edge server. Here are the
 units of measurement for each type of resource_name:
  o cpu_shares - CPU priority percentage
  o cpus - number of CPU cores
  o memory - amount of RAM in Mb
  o cpu_usage - CPU time in seconds
• cost - the total due for this resource
• id - Virtual machine ID

network_interfaces - an array of network interfaces used by this edge server 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:
WHITELIST IPS

- 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. Here are the units of measurement for each type of resource_name:
  - ip_addresses - number of IPs
  - rate - the port speed in Mb per second
  - data_received - amount of received data in Kb
  - data_sent - amount of sent data in Kb

- cost - the total due for the resource

Disks - an array of disks used by this edge 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. Here are the units of measurement for each type of resource_name:
    - disk_size - size in GB
    - data_read - read data in Kb
    - data_written - amount of written data in Kb
    - reads - number read operations
    - writes - number of write operations
  - cost - the total due for the resource

Edge_server - an array of edge server with its billing details:

- label - edge server name used in UI
- id - server ID used in database
- costs - an array of related resources with their total prices for the period specified in the stat-time parameter, where:
  - resource_name - the resource in question. In this case - template
  - value - here, the template ID in the database
  - cost - the total due for the resource

Total_cost - the total amount of money owed for the edge 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 edge server resources for the particular hour specified by stat_time parameter (memory, disks, templates) Usage_cost - the total due for
edge server usage for this particular hour specified by stat_time parameter (data sent/received, bandwidth, CPU usage).

## 24.26 Get CDN Edge Server CPU Usage Statistics

To view CPU usage statistics of a CDN edge server, run:

GET /edge_servers/:edge_server_id/cpu_usage.xml
GET /edge_servers/:edge_server_id/cpu_usage.json

Define a shorter period by setting Start and End time in the API call:


**XML Request example:**
```
curl -i GET -u user:userpass --url
http://onapp.test/edge_servers/:edge_server_id/cpu_usage.xml
```

**XML Request example:**
```
curl -i GET -u user:userpass --url
http://onapp.test/edge_servers/:edge_server_id/cpu_usage.json
```

Where you have to specify the edge server ID.

## 24.27 Search CDN Edge Server by Label

To search an edge server by label, run the following request

**XML Request example**
```
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:userpass --url
http://onapp.test/edge_servers.xml?q=label
```

**JSON Request example**

Where you have to specify the label of a CDN edge server you are searching for.
This chapter includes API request for managing HTTP caching rules. The HTTP Rules engine allows users to customize the CDN edge server behavior, e.g. how the CDN will manage cache and redirection.

A CDN resource can be configured with up to 100 rules. This is collectively called a ruleset. A rule consists of conditions and actions. A condition consists of a subject, which determines the value to select, and a predicate, which specifies what to compare the subject against. Conditions are bonded by the connectives "AND" or "OR". When all the conditions are met, the CDN edge server will perform the actions associated with the rule.

### 25.1 Get List of HTTP Caching Rules

To view the list of HTTP caching rules for a CDN resource, use the following request:

**GET /cdn_resources/:cdn_resource_id/http_caching_rules.xml**

**GET /cdn_resources/:cdn_resource_id/http_caching_rules.json**

**XML Request example:**

```bash
```

**JSON Request example:**

```bash
```

**XML Output example**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<object type="array">
  <object>
    <name>2</name>
    <conditions type="array">
      <condition>
        <subject>url</subject>
        <predicate>ends with</predicate>
        <value>gif</value>
      </condition>
    </conditions>
    <actions type="array">
      <action>
        <act>forbid client</act>
      </action>
    </actions>
  </object>
</object>
```
25.2 Add HTTP Caching Rule

To add an HTTP caching rule, use the following request:

POST /cdn_resources/:cdn_resource_id/http_caching_rules.xml
POST /cdn_resources/:cdn_resource_id/http_caching_rules.json

**XML Request example:**

```bash

'<rule><name>some_name</name><conditions><0><connective>if</connective><subject>url</subject><predicate>default</predicate><value></value></0><1><connective>and</connective><subject>cookie</subject><cookie></cookie><predicate>default</predicate><value></value></1><2><connective>and</connective><subject>param</subject><param></param><predicate>default</predicate><value></value></2><3><connective>and</connective><subject>header</subject><header></header><predicate>default</predicate><value></value></3></conditions><actions><0><act>force edge to cache</act><seconds></seconds></0><1><act>redirect client</act><url></url></1><2><act>set response header</act><header></header><value></value></2><3><act>set custom origin</act><value></value></3></actions></rule>'
```

**JSON Request example:**

```bash
curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' --url 'http://onapp.test/cdn_resources/:cdn_resource_id/http_caching_rules.json' -d '{"rule": {"name":"some_name", "conditions":{"0":{"connective":"if"}, "subject":"url", "predicate":"default", "value":""}, "1":{"connective":"and"}, ...
```

Where:
- **name** - the name of the rule
- **conditions** - the array of parameters of the conditions associated with the rule
- **subject** - the subject of the condition. For the list of subjects you can set for a rule refer to [The List of Subjects](#).
- **predicate** - the predicate of the condition. For the list of predicates you can set for a rule refer to [The List of Predicates](#).
- **value** - the value against which the subject is compared.
- **act** - the action associated with the rule. For the list of actions you can set for a rule refer to [The List of Actions](#).
OnApp 5.3 API Guide v2

WHERE:

name - name of the rule

conditions - the array of parameters of the conditions associated with the rule

connective - the connective by which the conditions are bonded, either 'and' or 'or'.

subject - the subject of the condition. The subject should be written using small letters only and with spaces between the words. For the list of subjects you can set for a rule refer to The List of Subjects.

predicate - the predicate of the condition. The predicate should be written using small letters only and with spaces between the words. For the list of predicates you can set for a rule refer to The List of Predicates.

value - the value against which the subject is compared.

header - the subject that selects the value of a specific client request header. If the request header does not exist, then the value "" is selected.

act - the action associated with the rule. The action should be written using small letters only and with spaces between the words. For the list of actions you can set for a rule refer to The List of Actions.

url - the subject that selects the URL part of the request. It excludes the query string.

cookie - the subject that selects the value of a specific cookie sent by the client.

param - the subject that selects the value of a specific query string parameter. If there are multiple identical keys, the last value is selected.

seconds - the time in seconds set for the action. For more information, refer to The List of Actions.

25.3 Edit HTTP Caching Rule

Use the following request to edit an HTTP caching rule:

PUT /cdn_resources/:cdn_resource_id/http_caching_rules.xml
PUT /cdn_resources/:cdn_resource_id/http_caching_rules.json

XML Request example

curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url 'http://onapp.test/cdn_resources/:cdn_resource_id/http_caching_rules.xml' -d '<id>1</id><rule><name>name</name><conditions><0><connective>if</connective></conditions>'

"subject":"cookie", "cookie":"", "predicate":"default", "value":""}, "2":{"connective":"and", "subject":"param", "param":"", "predicate":"default", "value":""}, "3":{"connective":"and", "subject":"header", "header":"", "predicate":"default", "value":""}, "actions":{"0":{"act":"force edge to cache", "seconds":""}, "1":{"act":"redirect client", "url":""}, "2":{"act":"set response header", "header":"", "value":""}, "3":{"act":"set custom origin", "value":""}}}}"
nnective><subject>url</subject><predicate>default</predicate><value></value></0><1><connective>and</connective><subject>cookie</subject><cookie></cookie><predicate>default</predicate><value></value></1><2><connective>and</connective><subject>param</subject><param></param><predicate>default</predicate><value></value></2><3><connective>and</connective><subject>header</subject><header></header><predicate>default</predicate><value></value></3></conditions><actions><0><act>force edge to cache</act><seconds></seconds></0><1><act>redirect client</act><url></url></1><2><act>set response header</act><header></header><value></value></2><3><act>set custom origin</act><value></value></3></actions></rule>

**JSON Request example**

```bash
curl -i -X PUT -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' --url 'http://onapp.test/cdn_resources/:cdn_resource_id/http_caching_rules.json' -d '{"id":1, "rule": {"name":"name", "conditions":{"0":{"connective":"if", "subject":"url", "predicate":"default", "value":""}, "1":{"connective":"and", "subject":"cookie", "cookie":"", "predicate":"default", "value":""}, "2":{"connective":"and", "subject":"param", "param":"", "predicate":"default", "value":""}, "3":{"connective":"and", "subject":"header", "header":"", "predicate":"default", "value":""}}, "actions":{"0":{"act":"force edge to cache", "seconds":""}, "1":{"act":"redirect client", "url":""}, "2":{"act":"set response header", "header":"", "value":""}, "3":{"act":"set custom origin", "value":""}}}'
```

Where:
- **id** - the ID of the HTTP caching rule you want to edit
- **name** - name of the rule
- **conditions** - the array of parameters of the conditions associated with the rule
- **connective** - the connective by which the conditions are bonded, either 'and' or 'or'.
- **subject** - the subject of the condition. The subject should be written using small letters only and with spaces between the words. For the list of subjects you can set for a rule refer to The List of Subjects.
- **predicate** - the predicate of the condition. The predicate should be written using small letters only and with spaces between the words. For the list of predicates you can set for a rule refer to The List of Predicates.
- **value** - the value against which the subject is compared.
- **cookie** - the subject that selects the value of a specific cookie sent by the client.
- **param** - the subject that selects the value of a specific query string parameter. If there are multiple identical keys, the last value is selected.
- **header** - the subject that selects the value of a specific client request header. If the request header does not exist, then the value "" is selected.
act - the action associated with the rule. The action should be written using small letters only and with spaces between the words. For the list of actions you can set for a rule refer to The list of Actions.

seconds - the time in seconds set for the action. For more information, refer to The list of Actions.

url - the subject that selects the URL part of the request. It excludes the query string.

25.4 Delete HTTP Caching Rule

Use the following request to delete an HTTP caching rule:

DELETE /cdn_resources/:cdn_resource_id/http_caching_rules/:rule_id.xml
DELETE /cdn_resources/:cdn_resource_id/http_caching_rules/:rule_id.json

XML Request example
curl -i -X DELETE -u user:userpass -H 'Accept: application/xml' --url
'http://onapp.test/cdn_resources/:cdn_resource_id/http_caching_rules/:rule_id.xml'

JSON Request example
curl -i -X DELETE -u user:userpass -H 'Accept: application/json' --url
'http://onapp.test/cdn_resources/:cdn_resource_id/http_caching_rules/:rule_id.json'
26 CDN REPORTING

CDN reporting functionality allows you to conduct the in-depth analysis of your own CDN resources by viewing different reports.

26.1 CDN Top Files Report

To get the Top 50 files report, use the following request:

GET /cdn/reports/top_files.xml
GET /cdn/reports/top_files.json

XML Request example

curl -i -X GET -u user:userpass --url http://onapp.test/cdn/reports/top_files.xml -d ' <top_files><start_date>2016-11-09</start_date><end_date>2016-11-10</end_date><entity_id>945986057</entity_id></top_files>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example


Where:
Define a time period by setting the following parameters:
- `start_date` - the start date of the specific time period
- `end_date` - the end date of the specific time period

Get statistics for a particular CDN resource by setting the following parameters:
- `entity_id` - the ID of the CDN resource

XML Output example:

<top_files>
<top_fifty_files_table type="array">
<top_fifty_files_table><resourceId type="integer">422564898</resourceId><fileUrl>/test.gif</fileUrl><request type="float">113.0</request>
</top_fifty_files_table>
</top_fifty_files_table>
</top_files>
<hit type="float">110.0</hit>
<miss type="float">3.0</miss>
<bANDwidth type="float">3781649.0</bandwidth>
</top_fifty_files_table>
</top_fifty_files_table>
</top_files>

Where:

resourceId - the ID of the CDN resource
fileUrl - the URL of the resource file
request - the total amount of file requests for the selected period
hit - the amount of successful file requests for the selected period
miss - the amount of failed file requests for the selected period
bandwidth - the amount of transmitted bandwidth for the selected period

26.2 Purge CDN Resource in Top Files Report

To remove content from cache, use the following request:
POST /cdn/reports/top_files/purge.xml
POST /cdn/reports/top_files/purge.json

XML Request example
  curl -i -X POST -u user:userpass
  http://onapp.test/cdn/reports/top_files/purge.xml -d
  '<root><remote_id>945986057</remote_id><url>/</url></root>'
  -H 'Accept:application/xml' -H 'Content-type:application/xml'

JSON Request example
  curl -i -X POST -u user:userpass
  http://onapp.test/cdn/reports/top_files/purge.json -d
  '{"remote_id": 945986057, "url": "/"}'
  -H 'Accept:application/json'
  -H 'Content-type:application/json'

Where:
remote_id - the ID of the CDN resource
url - the URL of the resource file
26.3 CDN Top Referrers Report

This report is available only for resource owner.

To get the Top 50 referrers report, use the following request:
GET /cdn/reports/top_referrers.xml
GET /cdn/reports/top_referrers.json

XML Request example
```
curl -i -X GET -u user:userpass --url http://onapp.test/cdn/reports/top_referrers.xml -d '  
  <top_referrers>
  <top_fifty_referrers_table type="array">
  <top_fifty_referrers_table>
  <resourceId type="integer">422564898</resourceId>
  <referrer>/test.gif</referrer>
  <hit type="float">110.0</hit>
  </top_fifty_referrers_table>
  </top_fifty_referrers_table>
  </top_referrers>
''

-H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request example
```
curl -i -X GET -u user:userpass --url http://onapp.test/cdn/reports/top_referrers.json -d 
'  {
  "top_referrers": {
    "start_date": "2016-11-09",
    "end_date": "2016-11-10",
    "entity_id": "945986057"
  }
  }' -H 'Accept: application/json' -
H 'Content-type: application/json'
```

Where:
Define a time period by setting the following parameters:
- **start_date** - the start date of the specific time period
- **end_date** - the end date of the specific time period

Get statistics for a particular CDN resource by setting the following parameters:
- **entity_id** - the ID of the CDN resource

XML Output example:
```
<top_referrers>
  <top_fifty_referrers_table type="array">
    <top_fifty_referrers_table>
      <resourceId type="integer">422564898</resourceId>
      <referrer>/test.gif</referrer>
      <hit type="float">110.0</hit>
    </top_fifty_referrers_table>
  </top_fifty_referrers_table>
</top_referrers>
```

Where:
**resourceId** - the ID of the CDN resource
**WHITELIST IPS** - **EDIT WHITELISTED IP**

- **referrer** - the referrer link
- **hit** - the amount of references for the selected period

## 26.4 CDN Overview Report

To get the Overview report, use the following request:

GET /cdn/reports/overview.xml
GET /cdn/reports/overview.json

**XML Request example**

curl -i -X GET -u user:userpass --url http://onapp.test/cdn/reports/overview.xml -d '<overview><frequency>2</frequency><start_date>2016-11-09</start_date><end_date>2016-11-10</end_date></overview>' -H 'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request example**

curl -i -X GET -u user:userpass --url http://onapp.test/cdn/reports/overview.json -d '{"overview": {"frequency":"2","start_date": "2016-11-09", "end_date": "2016-11-10"}}' -H 'Accept: application/json' -H 'Content-type: application/json'

**Where:**

Define statistics filters using the following parameters:

- **frequency** - set statistics frequency using the following values: 0- one minute, 1- one hour or 2- one day.
- **start_date** - the start date of the specific time period
- **end_date** - the end date of the specific time period

**XML Output example:**

```xml
<overview>
  <cache_statistic_line_chart type="array">
    <cache_statistic_line_chart>
      <time>2016-12-09</time>
      <cached_type="float">533803623.0</cached>
      <uncached_type="float">112393003.0</uncached>
      <hit_type="float">0.0</hit>
      <miss_type="float">0.0</miss>
      <gb_type="float">0.0</gb>
    </cache_statistic_line_chart>
  </cache_statistic_line_chart>
</overview>
```
<overview_top_five_http_error_codes_table type="array">
    <resourceId type="integer">741579723</resourceId>
    <errorRequest type="float">614.0</errorRequest>
</overview_top_five_http_error_codes_table>

<overview_top_five_resources_table type="array">
    <resourceId type="integer">741579723</resourceId>
    <bandwidth type="float">866516552.0</bandwidth>
    <cacheHit type="float">744.0</cacheHit>
    <miss type="float">508.0</miss>
</overview_top_five_resources_table>

<overview_top_five_visitor_locations_pie_chart type="array">
    <country>GB</country>
    <request type="float">1493.0</request>
</overview_top_five_visitor_locations_pie_chart>

Where:

cache_statistic_line_chart - an array of cache statistics chart details:
time - the selected time period
cached - the amount of cached bandwidth
uncached - the amount of uncached bandwidth
hit - the amount of successful file requests for the selected period
miss - the amount of failed file requests for the selected period
gb - the amount of transmitted bandwidth for the selected period

overview_top_five_http_error_codes_table - an array of top five error codes table details:
resourceld - the ID of CDN resource
errorRequest - the amount of error requests

overview_top_five_resources_table - an array of top five CDN resources table details:
resourceId - the ID of CDN resource
bandwidth - the amount of transmitted bandwidth

cacheHit - the amount of successful file requests
miss - the amount of failed file requests

overview_top_five_visitor_locations_pie_chart - an array of top five visitor locations pie chart details:
country - the country, where visitors are located
request - the amount of requests from the corresponding country
26.5 CDN Cache Statistics Report

To get the Cache Statistics report, use the following request:
GET /cdn/reports/cache_statistics.xml
GET /cdn/reports/cache_statistics.json

XML Request example
curl -i -X GET -u user:userpass --url
http://onapp.test/cdn/reports/cache_statistics.xml -d
'<!--cache_statistics><frequency>2</frequency><filter_type>1</filter_type><entity_id></entity_id><start_date>2016-11-09</start_date><end_date>2016-11-10</end_date></cache_statistics>'
-H 'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example
curl -i -X GET -u user:userpass --url
http://onapp.test/cdn/reports/cache_statistics.json -d
'{"cache_statistics":
  {"frequency":"2","filter_type":"1","entity_id":null,"start_date":
   "2016-11-09", "end_date": "2016-11-10"}}' -H 'Accept: application/json' -H 'Content-type: application/json'

Where:
Define statistics filters using the following parameters:
- frequency - set statistics frequency using the following values: 0 - one minute, 1 - one hour or 2 - one day
- filter_type - select the statistics type using the following values: 0 - GB, 1 - hit/miss or 2 - speed
- entity_id - set CDN resource ID, for which you want to view the statistics, or leave this parameter blank to choose all CDN resources
- start_date - the start date of the specific time period
- end_date - the end date of the specific time period

XML Output example:

<cache_statistics>
  <cache_statistic_line_chart type="array">
    <cache_statistic_line_chart>
      <time>2016-12-09</time>
      <cached type="float">533803623.0</cached>
      <uncached type="float">112393003.0</uncached>
      <hit type="float">0.0</hit>
      <miss type="float">0.0</miss>
      <gb type="float">0.0</gb>
    </cache_statistic_line_chart>
  </cache_statistic_line_chart>
</cache_statistics>
<cache_statistic_table type="array">
<cache_statistic_table>
<locationId type="integer">547</locationId>
<request type="float">1761.0</request>
<gb type="float">736003742.0</gb>
<hit type="float">0.0</hit>
<miss type="float">0.0</miss>
<speed type="float">0.0</speed>
</cache_statistic_table>
</cache_statistics>

Where:

cache_statistic_line_chart - an array of cache statistics chart details:
time - the selected time period
cached - the amount of cached bandwidth
uncached - the amount of uncached bandwidth
hit - the amount of successful file requests for the selected period
miss - the amount of failed file requests for the selected period
gb - the amount of transmitted bandwidth for the selected period

26.6 CDN Status Codes Report

To get the Status Codes report, use the following requests:

1. To get the breakdown of one or all publisher’s resource and total error code, use:
   
   GET /cdn/reports/status_codes.xml
   GET /cdn/reports/status_codes.json

2. To get the breakdown of location with total error code for selected resource, use:
   
   GET /cdn/reports/status_codes/error_requests.xml
   GET /cdn/reports/status_codes/error_requests.json
3. To get the breakdown of error code in graph and table for selected location, use:
   GET /cdn/reports/status_codes/location_error_requests.xml
   GET /cdn/reports/status_codes/location_error_requests.json

XML Request example
   curl -i -X GET -u user:userpass --url
   http://onapp.test/cdn/reports/status_codes.xml
   "<status_codes>
   <status_code_line_chart type="array">
   <status_code_line_chart>
   <time>2016-12-09</time>
   <statusCode>200</statusCode>
   <request type="float">183.0</request>
   </status_code_line_chart>
   <status_code_line_chart>
   <time>2016-12-09</time>
   <statusCode>403</statusCode>
   <request type="float">28.0</request>
   </status_code_line_chart>
   </status_code_line_chart>
   <status_code_table type="array">
   <status_code_table>
   <statusCode>200</statusCode>
   <request type="float">2962.0</request>
   </status_code_table>
   </status_code_table>
   </status_codes>

JSON Request example
   curl -i -X GET -u user:userpass --url
   http://onapp.test/cdn/reports/status_codes.json
   "{"status_codes": {"frequency": "2", "start_date": "2016-11-09", "end_date": "2016-11-10"}}" -H 'Accept: application/json' -H 'Content-type: application/json'

Where:
Define statistics filters using the following parameters:
- **frequency** - set statistics frequency using the following values: 0 - one minute, 1 - one hour or 2 - one day
- **entity_id** - set CDN resource ID, for which you want to view the statistics, or leave this parameter blank to choose all CDN resources
- **start_date** - the start date of the specific time period
- **end_date** - the end date of the specific time period

XML Output example:
<statusCode>403</statusCode>
<request type="float">1439.0</request>
</status_code_table>

http_error_code_table type="array">
  <resourceId type="integer">741579723</resourceId>
  <errorRequest type="float">1289.0</errorRequest>
</http_error_code_table>
</status_codes>

Where:
status_code_line_chart - an array of status code line chart details:
time - the selected time period
request - the amount of requests with the corresponding error
statusCode - the status code of an error

status_code_table - an array of status codes table details:
request - the amount of requests with the corresponding error
statusCode - the status code of an error

http_error_code_table - an array of http error code table details:
resourceId - the ID of CDN resource
errorRequest - the amount of error requests

26.7 CDN Bandwidth Statistics Report

To get bandwidth statistics report, use the following request:
GET /cdn/reports/bandwidth_statistics.xml
GET /cdn/reports/bandwidth_statistics.json

Be aware, that the bandwidth statistics report shows information on Stream type CDN resources only.

XML Request example
curl -i -X GET -u user:userpass
"http://onapp.test/cdn/reports/bandwidth_statistics.xml" -d
'"<bandwidth><type>MBPS</type><start>2016-12-12T19:52</start><end>2016-12-28T19:52</end><group_by>location</group_by><locations
type="array"><location>547</location></locations><resources type =
WHITELIST IPS - EDIT WHITELISTED IP

"array"<resource>898241191</resource></resources></bandwidth>''
-H 'Accept: application/xml' -H 'Content-type: application/xml'

**JSON Request example**

```bash
curl -i -X GET -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json'
"http://onapp.test/cdn/reports/bandwidth_statistics.json"
-d '{ "bandwidth": {"start": "2016-12-12T19:52","end": "2016-12-28T19:52", "locations": [547],"resources": [898241191],
"group_by":"location", "type": "MBPS"}}' -H 'Accept:
application/json' -H 'Content-type: application/json'
```

Where:

- **start** - the start date to generate statistics in the YYYY-MM-DD+hh:mm:ss format
- **end** - the end date to generate statistics in the YYYY-MM-DD+hh:mm:ss format
- **resources** - the identifier of the resource in Aflexi database. To get the identifier, check with cdn_reference parameter in the GET /cdn_resources/:id.{format} request
- **locations** - the ID of the location
- **type** - the statistics type (MBPS or GB). In MBPS mode you can get statistics for the last 10 days only. The older statistics is removed. There are no restrictions for GB mode.
- **group_by** - to get the bandwidth statistics breaking down per location or per resource via API, use the **group_by** parameter with two possible values: location and resource. In case **group_by = location** bandwidth stats is breaking down per location, and if **group_by = resource** is breaking down per resource. This is the optional parameter.

**XML output example if GB statistics type is selected**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<stats type="array">
  <stat>
    <date type="dateTime">2016-12-24T02:00:00+02:00</date>
    <locations type="array">
      <location>
        <547>
          <cached type="float">0.0</cached>
          <non_cached type="float">0.10429243099999999</non_cached>
        </547>
      </location>
    </locations>
  </stat>
</stats>
```

Where:

- **date** - the point of time for which the statistics is generated
- **location** - an array of locations with the following details per location:
  - **cached** - the amount of data cached
  - **non_cached** - the amount of content which is not cached
XML output example if MBPS statistics type is selected

```xml
<?xml version="1.0" encoding="UTF-8"?>
<stats type="array">
  <stat>
    <date type="dateTime">2016-12-23T02:00:00+02:00</date>
    <speed type="float">0.202613</speed>
  </stat>
  <stat>...</stat>
</stats>
```

Where:
- **date** - the point of time for which the statistics is generated
- **speed** - a bandwidth statistics speed (in Mbits/s)
CDN RESOURCES

A CDN resource is a host (e.g. a specific web server), the content of which you are going to distribute over the network of edge servers. The list of servers taking part in distributing/caching of data is limited to the locations added to those edge groups assigned to the resource.

27.1 Get List of CDN Resources

To see all CDN resources in the cloud, use the following request:
GET /cdn_resources.xml
GET /cdn_resources.json

XML Request example:
curl -i -X GET -u user:userpass http://onapp.test/cdn_resources.xml

JSON Request example:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources.json

To view the list of HTTP resources:

XML Request example:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources.xml?type=http

JSON Request example:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources.json?type=http

To view the list of VoD resources:

XML Request example:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources.xml?type=vod

JSON Request example:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources.json?type=vod

To view the list of live streaming resources:

XML Request example:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources.xml?type=streaming

JSON Request example:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources.json?type=streaming
http://onapp.test/cdn_resources.xml?type=live_streaming

**JSON Request example:**
```
curl -i -X GET -u user:userpass
```

**XML Output example**
```
<?xml version="1.0" encoding="UTF-8"?>
<cdn_resources type="array">
  <cdn_resource>  
    <cdn_hostname>cdn.1example.com</cdn_hostname>
    <cdn_ssl_certificate_id nil="true"/>
    <cname>990113320.r.worldcdn-beta.net</cname>
    <created_at type="datetime">2012-05-10T14:19:02+00:00</created_at>
    <id type="integer">2</id>
    <resource_type>HTTP_PULL</resource_type>
    <updated_at type="datetime">2012-05-10T14:19:02+00:00</updated_at>
    <user_id type="integer">1</user_id>
    <last_24h_cost type="float">0.0</last_24h_cost>
    <cname>990113320.r.worldcdn-beta.net</cname>
    <origins type="array">
      <origin>
        <value>1example.com</value>
      </origin>
    </origins>
  </cdn_resource>
  ...
</cdn_resources>
```

**Explanation of the data returned:**
- **cdn_hostname** - the hostname which will serve static content
- **cdn_ssl_certificate_id** - the ID of the custom SNI SSL certificate added to the resource
- **cname** - CNAME record
- **created_at** - the date when the CDN resource was created in the [YYYY][MM][DD][hh][mm][ss]Z format
- **id** - the resource ID in the database
- **resource_type** - CDN resource type
- **updated_at** - the date when the CDN resource was updated in the [YYYY][MM][DD][hh][mm][ss]Z format
- **user_id** - the ID of the user, who owns the resource
- **last_24h_cost** - the amount due for the last 24 hours
- **cdn_reference** - ID of the SSL certificate on the remote Aflexi server
- **origins** - an array of CDN origins with the following parameters:
- **origin** - the path from which the CDN requests the content
- **key** - access key, if any
value – the path to the content

27.2 Get CDN Resource Basic Details

To view details of the particular CDN resource:
GET /cdn_resources/:id.xml
GET /cdn_resources/:id.json

XML Output example:
```xml
<?xml version="1.0" encoding="UTF-8"?>
<cdn_resource>
  <cdn_hostname>cdn.test.com</cdn_hostname>
  <created_at type="datetime">2012-05-14T10:19:37+00:00</created_at>
  <id type="integer">3</id>
  <resource_type>HTTP_PULL</resource_type>
  <updated_at type="datetime">2012-05-14T10:19:37+00:00</updated_at>
  <user_id type="integer">1</user_id>
  <last_24h_cost type="float">0.0</last_24h_cost>
  <edge_groups type="array">
    <edge_group>
      <created_at type="datetime">2012-04-18T11:58:05+00:00</created_at>
      <id type="integer">158</id>
      <label>dfgfg</label>
    </edge_group>
  </edge_groups>
  <status>ACTIVE</status>
  <secondary_hostnames type="array"/>
  <ssl_on type="boolean">false</ssl_on>
  <ssl nil="true"/>
  <origins type="array">
    <origin>
      <key/></origin>
  </origins>
</cdn_resource>
```

Explanation of the data returned:
- **cdn_hostname** - the hostname which will serve static content
- **created_at** - the date when the resource was created
- **id** - the resource ID in the database
- **resource_type** - HTTP PULL or PUSH
- **updated_at** - the date when the resource was updated
- **user_id** - the ID of the user, who owns the resource
- **last_24h_cost** - the amount of money owed for the resource for the last 24 hours.
- **edge_groups** - the array of edge groups assigned to this resource, where:
created_at - the date when the edge group was created
label - the label of the particular edge group assigned
id - the edge group id
updated_at - the date when the edge group was updated
status - the resource status (can be Preparing, Active, Suspended)
secondary hostname - secondary CDN hostname
ssl_on - whether SSL is enabled for the resource or not
ssl - custom SNI SSL certificate if it is added to the resource
origins - the path from which the CDN requests the content
key - access key, if any
value - the path to the content

27.3 Get CDN Resource Advanced Details

To view advanced details of the CDN resource, use the following request:
GET /cdn_resources/:id/advanced.xml
GET /cdn_resources/:id/advanced.json

XML Output example
<?xml version="1.0" encoding="UTF-8"?>
<cdn_resource>
  <secondary_hostnames type="array">
    <secondary_hostname>test.com</secondary_hostname>
  </secondary_hostnames>
  <ip_access_policy>ALLOW_BY_DEFAULT</ip_access_policy>
  <ip_addresses/>
  <country_access_policy>BLOCK_BY_DEFAULT</country_access_policy>
  <countries type="array">
    <country>AL</country>
    <country>AR</country>
    <country>GT</country>
    <country>HR</country>
  </countries>
  <url_signing_on type="boolean">true</url_signing_on>
  <url_signing_key>dcahcqDAD</url_signing_key>
  <hotlink_policy>ALLOW_BY_DEFAULT</hotlink_policy>
  <domains>www.example.com</domains>
  <password_on type="boolean">true</password_on>
  <passwords>
    <password>
      <username2>password2</username2>
      <username1>password1</username1>
    </passwords>
  <password_unauthorized_html>password unauthorized</password_unauthorized_html>
  <flv_pseudo_on type="boolean">true</flv_pseudo_on>
  <mp4_pseudo_on type="boolean">true</mp4_pseudo_on>
  <limit_rate type="integer">80</limit_rate>
<limit_rate_after type="integer">13</limit_rate_after>
</cdn_resource>

Explanation of the data returned:

- secondary hostname – secondary CDN hostname
- ip_access_policy – displays access policy from a range of IP addresses; either NONE (disabled), ALLOW_BY_DEFAULT or BLOCKED_BY_DEFAULT
- country_access_policy – displays access policy to the CDN resource's content for specified countries; either NONE (disabled), ALLOW_BY_DEFAULT or BLOCKED_BY_DEFAULT
- url_signing_on – true if the access requires URL signing; otherwise false
- url_signing_key – the key for URL signing; a signed URL looks like:
  _http://example.com/filename?hash=url-signing-key=_
- hotlink_policy – displays the hotlink policy; either NONE (disabled), ALLOW_BY_DEFAULT or BLOCKED_BY_DEFAULT
- domains – domains related to hotlink_policy
- password_on – true, if the access to the resource is restricted; otherwise false
- passwords – an array of username and password for restricted access in the following format:
  <username>password</username>
- mp4_pseudo_on – 1 if MP4 pseudo streaming is enabled, otherwise set 0
- flv_pseudo_on – 1 if FLV pseudo streaming is enabled, otherwise set 0
- limit_rate – speed limit of a response to a client (per request) in KB/s. Maximum limit rate value - 2147483647 KB/s
- limit_rate_after – the amount after which the speed of a response to a client will be limited in KB. Maximum limit rate after value - 2147483647 KB

## 27.4 Add HTTP CDN Resource

To create an HTTP resource, use the following request:

POST /cdn_resources.xml
POST /cdn_resources.json

Add HTTP PULL XML Request example

curl -i -X POST -d
'<cdn_resource><cdn_hostname>cdn.test.co</cdn_hostname><cdn_ssl_certificate_id><cdn_ssl_certificate_id><edge_group_ids type="array"><edge_group_id type="integer">1</edge_group_id></edge_group_ids><resource_type>HTTP_PULL</resource_type><origin>test.origin.com</origin></cdn_resource>' -u user:userpass http://onapp.test/cdn_resources.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'

Add HTTP PULL JSON Request example

curl -i -u user:userpass -X POST
http://onapp.test/cdn_resources.json -H 'Accept: application/json'
WHITELIST IPS - EDIT WHITELISTED IP

- H 'Content-type: application/json' -d

  
  "cdn_resource":{
    "cdn_hostname":"cdn.test.co","resource_type":"HTTP_PULL","cdn_ssl_certificate_id":"ssl_cert_id","edge_group_ids":[1],
    "origin":"test.origin.com"}

Add HTTP PUSH XML Request example

  curl -i -X POST -d

  '<cdn_resource><cdn_hostname>cdn.test.co</cdn_hostname><cdn_ssl_certificate_id><cdn_ssl_certificate_id><edge_group_ids type="array"><edge_group_id type="integer">7</edge_group_id><resource_type="HTTP_PULL"><ftp_password>j3x8svqybA2gmrgYOZSrOlYToQ</ftp_password></cdn_resource>' -u user:userpass


Add HTTP PUSH JSON Request example

  curl -i -u user:userpass -X POST
  http://onapp.test/cdn_resources.json -H 'Accept: application/json'
  -H 'Content-type: application/json' -d

  
  "cdn_resource":{
    "cdn_hostname":"cdn.test.co","resource_type":"HTTP_PULL","cdn_ssl_certificate_id":"ssl_cert_id","edge_group_ids":[7],
    "ftp_password":"j3x8svqybA2gmrgYOZSrOlYToQ"}

Where:

- **origin** - the path from which the CDN requests the content. You can specify up to 3 origins. You can specify custom origin port (for HTTP pull resource only). To use the custom port for resource's origin, specify a port number using a colon (":"). For example, <origin>1.2.3.4:80</origin>

When you are specifying only one origin, it can be either a CDN hostname or an IP address. In case you are specifying more than one origin, they can only be IP addresses.

To send two or more origins in the API request, use array:

**XML example**

  <origins type="array">
    <origin>111.111.111.1</origin>
    <origin>111.111.111.2</origin>
  </origins>

**JSON example**

  
  
  "origins":["111.111.111.1", "222.222.22.222"]

**cdn_hostname** - indicate the hostname which will serve static content. Specify the following fourth-level domain name for this parameter to create a resource with SSL enabled: "example.r.worldssl.net", where replace the example with the desired name.

**resource_type** - HTTP_PULL or HTTP_PUSH
**WHITELIST IPS**

**EDIT WHITELISTED IP**

- **cdn_ssl_certificate_id** - the ID of the custom SNI SSL certificate you want to add to the resource. You should only specify the IDs of those certificates that were added by the user with whom the new resource will be associated.
- **cdn_ssl_certificate_id** - the ID of the custom SNI SSL certificate you want to add to the resource
- **edge_group_ids** * - indicate the ID(s) of required CDN edge groups
- **ftp_password** * - specify the FTP password if you add an HTTP PUSH CDN resource type. It can consist of 6-32 alphanumeric characters.

**Response**

You will get a response consisting of two parts - the header with HTTP status code and the response body including the parameters. At this stage some of the parameters can be empty. This is expected behavior, because the full process of creation takes some time, and remote service cannot fill in the parameters at this time. For the complete list of parameters use GET request.

### 27.4.1 Page history

**OnApp 4.0:**
- **origin** parameter allows adding port to HTTP pull resource.
- Added **cdn_ssl_certificate_id** parameter that enables a user to associate a custom SNI SSL certificate with a CDN resource

### 27.5 Add Video on Demand CDN Resource

To create an HTTP resource, use the following request:

**POST /cdn_resources.xml**

**POST /cdn_resources.json**

**Add VoD PULL XML Request example**

curl -i -X POST -d
'<cdn_resource><cdn_hostname>az.test.api</cdn_hostname><edge_group_ids type="array"><edge_group_id type="integer">7</edge_group_id></edge_group_ids><resource_type>STREAM_VOD_PULL</resource_type><origin>test.origin.com</origin><cdn_resource>' -u user:pass http://onapp.test/cdn_resources.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'

**Add VoD PULL JSON Request example**

curl -i -u user:pass -X POST
'{"cdn_resource":{"cdn_hostname":"cdn.test.co","resource_type":"STREAM_VOD_PULL","edge_group_ids":[7],"origin":"test.origin.com"}}'
Add VoD PUSH XML Request example

```bash
curl -i -X POST -d
'<!cdn_resource><cdn_hostname>cdn.test.co</cdn_hostname><edge_group_ids type="array"><edge_group_id type="integer">7</edge_group_id></edge_group_ids><resource_type>STREAM_VOD_PUSH</resource_type><ftp_password>j3x8svqybA2gmrzY0Zs01YToQ</ftp_password></cdn_resource>' -u user:userpass
```

Add VoD PUSH JSON Request example

```bash
curl -i -u user:userpass -X POST
'{"cdn_resource":{"cdn_hostname":"cdn.test.co","resource_type":"STREAM_VOD_PUSH","edge_group_ids":[7],"ftp_password":"j3x8svqybA2gmrzY0Zs01YToQ"}}'
```

Where:
- **cdn_hostname** * - specify the name which will serve as a label only
- **resource_type** * - specify the resource type - STREAM_VOD_PULL or STREAM_VOD_PUSH
- **origin** * - the path from which the CDN requests the content (for VoD PULL request).
- **edge_group_ids** * - indicate the ID(s) of required CDN edge groups
- **ftp_password** * - specify the FTP password for VoD PUSH type. It should consist of 6-32 alphanumeric symbols.

Response
You will get a response consisting of two parts - the header with **HTTP status code** and the response body including the parameters. At this stage some of the parameters can be empty. This is expected behavior, because the full process of creation takes some time, and remote service cannot fill in the parameters at this time. For the complete list of parameters use GET request.

**27.6 Add Live Streaming CDN Resource**

To create a live streaming CDN resource, use the following request:

POST /cdn_resources.xml
POST /cdn_resources.json

**XML Request example (with external publishing point)**

```bash
curl -i -X POST -d
'<!cdn_resource><cdn_hostname>onapp.stream.resource</cdn_hostname><edge_group_ids type="array"><edge_group_id type="integer">1</edge_group_id></edge_group_ids><resource_type>STREAM_LIVE</resource_type><publishing_point>external</publishing_point><publishing_location>www.google.com</publishing_location><failove
```
WHITELIST IPS - EDIT WHITELISTED IP


JSON Request example (with external publishing point)

```
```

XML Request example (with internal publishing point)

```
curl -i -X POST -d '<cdn_resource><cdn_hostname>onapp.stream.resource</cdn_hostname><edge_group_ids type="array"><edge_group_id type="integer">1</edge_group_id></edge_group_ids><resource_type>STREAM_LIVE</resource_type><publishing_point>internal</publishing_point><publishing_location>532</publishing_location><failover_publishing_location>128</failover_publishing_location></cdn_resource>' -u user:userpass http://onapp.test/cdn_resources.xml -H 'Accept: application/xml' -H 'Content-type: application/xml'
```

JSON Request example (with internal publishing point)

```
curl -i -u user:userpass -X POST http://test/cdn_resources.json -H 'Accept: application/json' -H 'Content-type: application/json' -d '{"cdn_resource":{"cdn_hostname":"onapp.stream.resource","resource_type":"STREAM_LIVE","edge_group_ids":[7],"publishing_point":"internal","publishing_location":"532","failover_publishing_location":"128"}}'
```

Where:
- `cdn_hostname`* - specify the name which will serve as a label only
- `resource_type`* - STREAM_LIVE
- `publishing_point`* - the publishing point type: external or internal
- `publishing_location`* - specify the URL address for external publishing point. Set the ID of a location that will serve as a publishing point for internal type.
- `failover_publishing_location` - publishing point failover URL for external publishing point. Specify the ID of a location that will serve as a failover publishing point for internal type.
- `edge_group_ids`* - indicate the ID(s) of required CDN edge groups
Response
You will get a response consisting of two parts - the header with HTTP status code (see Introduction for details) and the response body including the parameters. At this stage some of the parameters can be empty. This is expected behavior, because the full process of creation takes some time, and remote service cannot fill in the parameters at this time. For the complete list of parameters use GET request.

27.7 Add HTTP Pull CDN Resource with Advanced Settings

To create an HTTP resource with advanced settings, use the following request:
POST /cdn_resources.xml
POST /cdn_resources.json
XML Request example

```
  '<cdn_resource><resource_type>HTTP_PULL</resource_type><cdn_hostname>HTTPPULCDNresource.com</cdn_hostname><origin>109.123.105.178</origin><cdn_ssl_certificate_id>ssl_sert_id</cdn_ssl_certificate_id><edge_group_ids type="array"><edge_group_id>225</edge_group_id></edge_group_ids><secondary_hostnames type="array"><secondary_hostname>test110.com</secondary_hostname><secondary_hostname>test220.com</secondary_hostname><ip_access_policy>BLOCK_BY_DEFAULT</ip_access_policy><ip_addresses>111.111.11.111,222.222.22.222</ip_addresses><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><countries type="array"><country>AL</country><country>GT</country></countries><hotlink_policy>ALLOW_BY_DEFAULT</hotlink_policy><domains>abuse.co.ua</domains><url_signing_on>1</url_signing_on><url_signing_key>12345qwertyyu</url_signing_key><cache_expiry>45</cache_expiry><password_on>1</password_on><form_pass><user type="array"><string>user123new</string><string>user234new</string></user><pass type="array"><string>passw123new</string><string>passw234new</string></pass></form_pass><password_unauthorized_html>YOU ARE NOT AUTHORIZED</password_unauthorized_html><flv_pseudo_on>1</flv_pseudo_on><mp4_pseudo_on>0</mp4_pseudo_on><limit_rate>150</limit_rate><limit_rate_after>1</limit_rate_after><proxy_cache_key>$host$uri</proxy_cache_key'>$host$uri</proxy_cache_key>
```
**JSON Request example**

```bash
```

Where:

- **origin** - the path from which the CDN requests the content. When using the hostnames according to RFC 1035, the origin may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], and dash [-]. The limit for hostname is 255 chars. You can specify up to 3 origins.

  - When you are specifying only one origin, it can be either a CDN hostname or an IP address. In case you are specifying more than one origin, they can only be IP addresses. To send two or more origins in the API request, use array. For example:

    ```json
    <origins type="array">
      <origin>111.111.111.111</origin>
      <origin>111.111.111.111</origin>
    </origins>
    ```

- **cdn_hostname** - indicate the hostname which will serve static content. Specify the following fourth-level domain name for this parameter to create a resource with SSL enabled: "example.r.worldssl.net", where replace the example with the desired name.

  - **ssl_on** - set to 'true' to enable SSL, otherwise set to 'false'. See the note below for more information.

**SSL**

- If the CDN hostname ends with ".r.worldssl.net", SSL will be enabled automatically.

  - To disable, remove the ".r.worldssl.net" ending and send the "ssl_on":false parameter. To enable, add the ".r.worldssl.net" ending to the cdn_hostname and send the "ssl_on":true parameter.

"y_cache_key"<proxy_read_time_out>60</proxy_read_time_out><proxy_connect_time_out>20</proxy_connect_time_out><http_bot_blocked>1</http_bot_blocked><origin_policy>HTTP</origin_policy></cdn_resource>"
Be aware that if CDN hostname ends with 'r.worldssl.net', it can not be digit-only (for example 123456.r.worldssl.net is not applicable).

**resource_type** - HTTP_PULL

**cdn_ssl_certificate_id** - the ID of the custom SNI SSL certificate you want to add to the resource. You should only specify the IDs of those certificates that were added by the user with whom the new resource will be associated.

**edge_group_ids** - indicate the ID(s) of required CDN edge groups

**secondary_hostnames** - an array of secondary CDN hostnames. You can add up to 7 secondary CDN hostnames. For example:

```xml
<secondary_hostnames type="array">
  <secondary_hostname>test100.com</secondary_hostname>
  <secondary_hostname>test200.com</secondary_hostname>
</secondary_hostnames>
```

To be able to use a secondary hostname for the CDN resource with SSL enabled, you require an SSL certificate for your custom hostname. For help with questions about the SSL certificate purchase, please contact OnApp support.

**ip_access_policy** - configure a rule to control access to the CDN resource's content for a range of IP addresses:

- **ALLOW_BY_DEFAULT** - allow IP access policy by default, except for IP addresses specified in the **ip_addresses** parameter
- **BLOCK_BY_DEFAULT** - block IP access policy by default, except for IP addresses specified in the **ip_addresses** parameter
- **NONE** - switch off the IP access policy

**ip_addresses** - IP address(es) related to **ip_access_policy** parameter; The comma-separated list of IP addresses or IP ranges allowed/blocked by default. Use the following format "10.10.10.10, 20.20.20.0/24"

**hotlink_policy** - configure hotlink policy properties to protect your content from unauthorized hotlinking:

- **ALLOW_BY_DEFAULT** - allow hotlink policy by default, except for domains specified in the **domains** parameter
- **BLOCK_BY_DEFAULT** - block hotlink policy by default, except for domains specified in the **domains** parameter
- **NONE** - switch off the rule

**domains** - domains related to **hotlink_policy**

**hls_on** - set to 1 to enable HTTP Live Streaming (HLS) Optimization

**hls_force_cache** - set to 1 to create an HTTP rule that will enforce cache expiry. This option is only available when the **hls_on** parameter is set to 1

**country_access_policy** - configure a rule to control access to the CDN resource's content for specified countries:

- **ALLOW_BY_DEFAULT** - allow country access policy by default, except for countries specified in the **countries** parameter
• **BLOCK_BY_DEFAULT** - block country access policy by default, except for countries specified in the *countries* parameter

• **NONE** - switch off the country access policy

*countries* - country codes, related to *country_access_policy* in ISO 3166-1 alpha-2 format

*cache_expiry* - set the cache expiry time in minutes (min=1, max=35000000)

*url_signing_on* - set 1 to enable and protect your files from unauthorized access with a key

*url_signing_key* - input the key for URL signing. Input letters and digits (6-32 symbols).

*password_on* - set 1 to enable and to restrict access to the resource (*cdn_hostname*), otherwise set 0

*form_pass* - an array with usernames and passwords to access the resource

*pass* - the user password.

*user* - the user login, which may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], underscore [ _ ], and dash [ - ].The first symbol should be alphabetic. The username cannot be duplicated.

*password_unauthorized_html* - text, which will be displayed in case of fail of authentication. Max 1000 chars.

*mp4_pseudo_on* - set 1 to enable MP4 pseudo streaming, otherwise set 0

*flv_pseudo_on* - set 1 to enable FLV pseudo streaming, otherwise set 0

*ignore_set_cookie_on* - set 1 to enable caching content with Set-Cookie response headers, otherwise set 0 to ignore content caching

**Nginx Settings**

• **limit_rate** - sets speed limit of a response to a client (per request) in KB/s. Maximum limit rate value - 2147483647 KB/s

• **limit_rate_after** - sets the amount after which the speed of a response to a client will be limited in KB. Maximum limit rate after value -2147483647 KB

• **proxy_read_time_out** - proxy server response timeout in seconds. Maximum proxy read timeout value - 65535 seconds

• **proxy_connect_time_out** - timeout for establishing connection with proxy server in seconds. Maximum proxy connect time out value - 75 seconds.

• **proxy_cache_key** - specify the key for caching. This parameter defines what information is included in the cache key. You can set the following options:
  
  o $host$request_uri
  
  o $host$uri
  
  o $proxy_host$request_uri
  
  o $proxy_host$uri

*http_bot_blocked* - set 1 to block web crawling bots from indexing the CDN content (for HTTP Pull CDN resources only)
**origin_policy** - set the parameter to choose the type of connection. Possible values are: HTTP, HTTPS, AUTO.

**Response**

You will get a response consisting of two parts - the header with HTTP status code and the response body including the parameters. At this stage some of the parameters can be empty. This is expected behavior, because the full process of creation takes some time, and remote service cannot fill in the parameters at this time. For the complete list of parameters use GET request.

**XML Response Example**

```
HTTP/1.1 201
Date: Wed, 31 Jul 2013 09:19:55 GMT
Server: Apache/2.2.3 (CentOS)
X-Powered-By: Phusion Passenger (mod_rails/mod_rack) 3.0.17
X-UA-Compatible: IE=Edge,chrome=1
ETag: "915066feccf0b14edaa50af485b3b705"
Cache-Control: max-age=0, private, must-revalidate
X-Request-Id: f48bc47604a5784aaf5f98dab733465
X-Runtime: 0.387472
X-Rack-Cache: invalidate, pass
Set-Cookie: _session_id=2cc141e9002d8d3df5c4033dbda2a090; path=/; HttpOnly
Location: http://onapp.test/cdn_resources/10743
Status: 201
Content-Length: 797
Connection: close
Content-Type: application/xml; charset=utf-8

<?xml version="1.0" encoding="UTF-8"?>
<cdn_resource>
  <cdn_hostname>PI-HTTPPULCDNresource.com</cdn_hostname>
  <cname></cname>
  <created_at type="datetime">2013-07-31T12:19:55+03:00</created_at>
  <id type="integer">10743</id>
  <resource_type>HTTP_PULL</resource_type>
  <updated_at type="datetime">2013-07-31T12:19:55+03:00</updated_at>
  <user_id type="integer">20</user_id>
  <last_24h_cost type="float">0.0</last_24h_cost>
  <cname></cname>
  <cdn_reference type="integer">431059243</cdn_reference>
  <secondary_hostnames type="array">
    <secondary_hostname>test100.com</secondary_hostname>
  </secondary_hostnames>
</cdn_resource>
```
WHITELIST IPS

<secondary_hostname>test200.com</secondary_hostname>
</secondary_hostnames>
<origins type="array">
  <origin>111.111.111.1</origin>
</origins>
<ssl_on nil="true"/>
</cdn_resource>

27.7.1 Page history

v 5.1

- added the hls_force_cache parameter
- added the hls_on parameter

v 4.0

- Added cdn_ssl_certificate_id parameter that enables a user to associate a custom SNI SSL certificate with a CDN resource

v. 3.3.1

- Added the following parameter:
  - origin_policy

If the CDN hostname ends with '.r.worldssl.net', SSL will be enabled automatically.

To disable, remove the '.r.worldssl.net' ending and send the "ssl_on":false parameter. To enable, add the '.r.worldssl.net' ending to the cdn_hostname and send the "ssl_on":true parameter.

Be aware that if CDN hostname ends with '.r.worldssl.net', it can not be digit-only (for example 123456.r.worldssl.net is not applicable).

27.8 Add HTTP Push CDN Resource with Advanced Settings

To create an HTTP resource with advanced settings, use the following request:

POST /cdn_resources.xml
POST /cdn_resources.json

XML Request example

curl -i -X POST -u user:userpass
OnApp 5.3 API Guide

WhiteList IPS - Edit Whitelisted IP

HTTP: onapp.test/cdn_resources.xml -H 'Accept: application/xml' -H 'Content-type: application/xml' -d

'cdn_resource'<resource_type>HTTP_PUSH</resource_type><cdn_hostname>xmlcdn.apitest.com</cdn_hostname><cdn_ssl_certificate_id>ssl_sert_id</cdn_ssl_certificate_id><edge_group_ids type="array">edge_group_id type="integer">225</edge_group_ids><storage_server_location>816382921</storage_server_location><ftp_password>qwerty123

JSON Request example


"cdn_resource":{"resource_type":"HTTP_PUSH","cdn_hostname":"jsoncdn.apitest.com","cdn_ssl_certificate_id":"ssl_sert_id","edge_group_ids":"[225]","storage_server_location":"816382921","ftp_password":"qwerty123","secondary_hostnames":["test122.com","test234.com"],"ip_access_policy":"ALLOW_BY_DEFAULT","ip_addresses":"111.111.111.111,222.222.22.222","country_access_policy":"BLOCK_BY_DEFAULT","countries":["AL","AR","GT","HR"],"hotlink_policy":"ALLOW_BY_DEFAULT","domains":"mnw.netggl.com","url_signing_on":1,"url_signing_key":"12345qwqew","password_on":1,"form_pass":{"user":["user190","user278"],"pass":["pass123","pass2348"],"password_unauthorized_html":"YOU ARE NOT AUTHORIZED"},"mp4_pseudo_on":1,"flv_pseudo_on":1,"limit_rate":150,"limit_rate_after":1"}

Where:

- cdn_hostname* - indicate the hostname which will serve static content. Specify the following fourth-level domain name for this parameter to create a resource with
SSL enabled: "example.r.worldssl.net", where replace the example with the desired name.

ssl_on - set to 'true' to enable SSL, otherwise set to 'false'. See the note below for more information.

**SSL**

If the CDN hostname ends with '.r.worldssl.net', SSL will be enabled automatically.

To disable, remove the '.r.worldssl.net' ending and send the "ssl_on":false parameter. To enable, add the '.r.worldssl.net' ending to the cdn_hostname and send the "ssl_on":true parameter.

Be aware that if CDN hostname ends with '.r.worldssl.net', it can not be digit-only (for example 123456.r.worldssl.net is not applicable).

`resource_type`* - HTTP_PUSH

`cdn_ssl_certificate_id` - the ID of the custom SNI SSL certificate you want to add to the resource. You should only specify the IDs of those certificates that were added by the user with whom the new resource will be associated.

`edge_group_ids`* - indicate the ID(s) of required CDN edge groups

`storage_server_location` - the ID of the storage server location which should be assigned to this resource. To get the ID of the required storage server location, use the request described at the GetList of Available Storage Locations section. If no location set, the first active storage server is chosen automatically.

`ftp_password`* - indicate the FTP server password. It should consist of 6-32 alphanumeric symbols.

`secondary_hostnames` - an array of secondary CDN hostnames. You can add up to 7 secondary CDN hostnames.

```
<secondary_hostnames type="array">
  <secondary_hostname>test100.com</secondary_hostname>
  <secondary_hostname>test200.com</secondary_hostname>
</secondary_hostnames>
```

To be able to use a secondary hostname for the CDN resource with SSL enabled, you require an SSL certificate for your custom hostname. For help with questions about the SSL certificate purchase, please contact OnApp support.

`ip_access_policy` - configure a rule to control access to the CDN resource's content for a range of IP addresses:
- **ALLOW_BY_DEFAULT** - allow IP access policy by default, except for IP addresses specified in the `ip_addresses` parameter
- **BLOCK_BY_DEFAULT** - block IP access policy by default, except for IP addresses specified in the `ip_addresses` parameter
- **NONE** - switch off the IP access policy

`ip_addresses` - IP address(es) related to `ip_access_policy` parameter; The comma-separated list of IP addresses or IP ranges allowed/blocked by default. Use the following format "10.10.10.10, 20.20.20.0/24"

`hotlink_policy` - configure hotlink policy properties to protect your content from unauthorized hotlinking:
- **ALLOW_BY_DEFAULT** - allow hotlink policy by default, except for domains specified in the `domains` parameter
WHITELIST IPS - EDIT WHITELISTED IP

- BLOCK_BY_DEFAULT - block hotlink_policy by default, except for domains specified in the domains parameter
- NONE - switch off the rule
domains - domains related to hotlink_policy
country_access_policy - configure a rule to control access to the CDN resource's content for specified countries:
- ALLOW_BY_DEFAULT - allow country access policy by default, except for countries specified in the countries parameter
- BLOCK_BY_DEFAULT - block country access policy by default, except for countries specified in the countries parameter
- NONE - switch off the country access policy
countries - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.
url_signing_on - set 1 to enable and protect your files from unauthorized access with a key
url_signing_key - input the key for URL signing. Input letters and digits (6-32 symbols).
form_pass - an array with usernames and passwords to access the resource:
  pass - the user password.
  user - the user login, which may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], underscore [ _ ], and dash [ - ]. The first symbol should be alphabetic. The username cannot be duplicated.
password_unauthorized_html - text, which will be displayed in case of fail of authentication. Max 1000 chars.
mp4_pseudo_on - set 1 to enable MP4 preudo streaming, otherwise set 0
flv_pseudo_on - set 1 to enable FLV preudo streaming, otherwise set 0

Nginx Settings
- limit_rate - sets speed limit of a response to a client (per request) in KB/s. Maximum limit rate value - 2147483647 KB/s
- limit_rate_after - sets the amount after which the speed of a response to a client will be limited in KB. Maximum limit rate after value -2147483647 KB

Page History
v. 4.0
- Updated cdn_hostname parameter with ability to enable SSL

- Added cdn_ssl_certificate_id parameter that enables a user to associate a custom SNI SSL certificate with a CDN resource

v. 3.1.1
- Added `storage_server_location` parameter.

https://docs.onapp.com/display/40API/Add+HTTP+Push+CDN+Resource+with+Advanced+Settings

27.9 Add VoD Push CDN Resource With Advanced Settings

To create new video on demand resource with advanced settings:

POST /cdn_resources.xml
POST /cdn_resources.json

**XML Request example**

```
  <cdn_resource><resource_type>STREAM_VOD_PUSH</resource_type><cdn_hostname>apitest.com</cdn_hostname><ftp_password>testpassword</ftp_password><edge_group_ids type="array"><edge_group_id>1</edge_group_id><edge_group_id>2</edge_group_id><hotlink_policy>BLOCK_BY_DEFAULT</hotlink_policy><domains>goo.coabuse.ua</domains><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><countries type="array"><country>AL</country><country>GT</country></countries><secure_wowza_on>1</secure_wowza_on><secure_wowza_token>test123456</secure_wowza_token><storage_server_location>4637643278</storage_server_location><token_auth_on>1</token_auth_on><token_auth_secure_paths type="array"><token_auth_secure_path>/Video1</token_auth_secure_path><token_auth_secure_path>/Video2</token_auth_secure_path></token_auth_secure_paths><token_auth_backup_key>fgff45788787878</token_auth_backup_key><token_auth_secure_paths type="array"><token_auth_secure_path>/Video1</token_auth_secure_path><token_auth_secure_path>/Video2</token_auth_secure_path></token_auth_secure_paths></cdn_resource>'
```

**JSON Request example**

```
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/cdn_resources.json -d '{"cdn_resource":{"resource_type":"STREAM_VOD_PUSH","ftp_password":"testpassword","cdn_hostname":"apitest.com","edge_group_ids":["1","2"],"hotlink_policy":"BLOCK_BY_DEFAULT","domains":"test.com","country_access_policy":"ALLOW_BY_DEFAULT","countries":["AL","GT"],"storage_server_location":"4545566","secure_wowza_on":"1","secure_wowza_token":"test123456","token_auth_on":"1","token_auth_security_paths":null}'
```
Where:

- **cdn_hostname** - specify the name which will serve as a label only
- **resource_type** - STREAM_VOD_PUSH
- **ftp_password** - indicate the FTP server password. It should consist of 6-32 alphanumeric symbols.
- **edge_group_ids** - indicate the ID(s) of required CDN edge groups
- **advanced_settings** - set 1 to enable advanced settings
- **hotlink_policy** - configure hotlink policy properties to protect your content from unauthorized hotlinking:
  - BLOCK_BY_DEFAULT - block hotlink_policy by default, except for domains specified in the domains parameter
  - NONE - switch off the rule
- **domains** - domains related to hotlink_policy
- **country_access_policy** - configure a rule to control access to the CDN resource’s content for specified countries:
  - BLOCK_BY_DEFAULT - block country access policy by default, except for countries specified in the countries parameter
  - NONE - switch off the country access policy
- **countries** - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.
- **secure_wowza_on** - set 1 to enable secure Wowza streaming encryption, otherwise set 0
- **secure_wowza_token** - specify the Wowza token
- **token_auth_on** - indicate whether Token Authentication is enabled for this resource. It's only supported for VoD PUSH, VoD PULL, and Live Streaming resource.
- **token_auth_primary_key** - set primary key to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters. This must not be blank if token_auth_on is enabled.
- **token_auth_backup_key** - set backup key (optional) to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt (if primary key failed) the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters.
- **token_auth_secure_paths** - set secure paths that mark streaming to these paths requires a valid token to play a stream. Include only the relative path that appears after the content access point. If undefined, forward slash (/) will be set, means the resource is secured at root level and all streaming request to the resource will be granted only if the provided token is valid. Example - ["/video1", "/video2"]
- **storage_server_location** - the ID of the storage server location which should be assigned to this resource. To get the ID of the required storage server location, use the request described at the Get List of Available Storage Locations section. If no location set, the first active storage server is chosen automatically.
Response
You will get a response consisting of two parts - the header with HTTP status code and the response body including the parameters. At this stage some of the parameters can be empty. This is expected behavior, because the full process of creation takes some time, and remote service cannot fill in the parameters at this time. For the complete list of parameters use GET request.

Page History
v.3.3.1 adds the following parameters:

- token_auth_on
- token_auth_primary_key
- token_auth_backup_key
- token_auth_secure_paths
- storage_server_location

27.10 Add VoD Pull CDN Resource With Advanced Settings

To create new video on demand resource with advanced settings:

POST /cdn_resources.xml
POST /cdn_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/cdn_resources.xml -d "<cdn_resource><resource_type>STREAM_VOD_PULL</resource_type><cdn_hostname>apitest.com.ua</cdn_hostname><origin>111.111.11.111</origin><edge_group_ids type="array"/><edge_group_id>1</edge_group_id><edge_group_id>2</edge_group_id><hotlink_policy>BLOCK_BY_DEFAULT</hotlink_policy><domains>goo.coabuse.ua</domains><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><countries type="array"/><country>AL</country><country>GT</country><secure_wowza_on>1</secure_wowza_on><secure_wowza_token>test123456</secure_wowza_token><token_auth_on>1</token_auth_on><token_auth_primary_key>zsfdfasga</token_auth_primary_key><token_auth_secure_paths type="array"/><token_auth_secure_path>/Video1</token_auth_secure_path>"
h><token_auth_secure_path>/Video2</token_auth_secure_path></token_auth_secure_paths><token_auth_backup_key>fgff45788787878</token_auth_backup_key><token_auth_secure_paths type="array"><token_auth_secure_path>/Video1</token_auth_secure_path><token_auth_secure_paths><token_auth_secure_path>/Video2</token_auth_secure_path></token_auth_secure_paths></cdn_resource>'

JSON Request example

curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/cdn_resources.json -d '{"cdn_resource":{"resource_type":"STREAM_VOD_PULL","origin":"111.111.11.111","cdn_hostname":"apitest.com","edge_group_ids": ["1","2"],"hotlink_policy":"BLOCK_BY_DEFAULT","domains": "test.com","country_access_policy":"ALLOW_BY_DEFAULT","countries": ["AL","GT"],"secure_wowza_on": "1", "secure_wowza_token": "test123456","token_auth_on": "1","token_auth_primary_key": "zsfdfasga","token_auth_secure_paths": ["/video1","/video2"],"token_auth_backup_key": "fgff45788787878"}}'

Where:

cdn_hostname* - specify the name which will serve as a label only
resource_type* - STREAM_VOD_PULL
origin* - the path from which the CDN requests the content. When using the hostnames according to RFC 1035, the origin may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], and dash [-]. The limit for hostname is 255 chars. The VoD Pull resource can have only one origin.
edge_group_ids* - indicate the ID(s) of required CDN edge groups
advanced_settings* - set 1 to enable advanced settings
hotlink_policy - configure hotlink policy properties to protect your content from unauthorized hotlinking:
  • BLOCK_BY_DEFAULT - block hotlink_policy by default, except for domains specified in the domains parameter
  • NONE - switch off the rule
domains - domains related to hotlink_policy
country_access_policy - configure a rule to control access to the CDN resource's content for specified countries:
  • BLOCK_BY_DEFAULT - block country access policy by default, except for countries specified in the countries parameter
  • NONE - switch off the country access policy
countries - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.
secure_wowza_on - set 1 to enable secure Wowza streaming encryption, otherwise set 0
secure_wowza_token - specify the Wowza token
token_auth_on - indicate whether Token Authentication is enabled for this resource. It's only supported for VoD PUSH, VoD PULL, and Live Streaming resource.
token_auth_primary_key - set primary key to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters. This must not be blank if token_auth_on is enabled.

token_auth_backup_key - set backup key (optional) to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt (if primary key failed) the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters.

token_auth_secure_paths - set secure paths that marks streaming to these paths requires a valid token to play a stream. Include only the relative path that appears after the content access point. If undefined, forward slash (/) will be set, means the resource is secured at root level and all streaming request to the resource will be granted only if the provided token is valid. Example - ["/video1", "/video2"]

Response
You will get a response consisting of two parts - the header with HTTP status code (see Introduction for details) and the response body including the parameters. At this stage some of the parameters can be empty. This is expected behavior, because the full process of creation takes some time, and remote service cannot fill in the parameters at this time. For the complete list of parameters use GET request.

Page History
v.3.3.1 adds the following parameters:

- token_auth_on
- token_auth_primary_key
- token_auth_backup_key
- token_auth_secure_paths
- storage_server_location

27.11 Add Live Streaming CDN Resource with Advanced Settings

POST /cdn_resources.xml
POST /cdn_resources.json

XML Request example (with internal publishing point):
'<cdn_resource><cdn_hostname>testLV.internal</cdn_hostname><resource_type>STREAM_LIVE</resource_type><publishing_point>internal</publishing_point><publishing_location>532</publishing_location><failover_pUBLISHING_LOCATION>336</failover_pUBLISHING_LOCATION><hotlink_policy>BLOCK_BY_DEFAULT</hotlink_policy><domains>www.google.com</domains><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><countries>AO</countries><countries>BH</countries><secure_wowza_on>1</secure_wowza_on><secure_wowza_token>4t534564tyrt</secure_wowza_token><edge_group_ids>224</edge_group_ids><token_auth_on>1</token_auth_on><token_auth_primary_key>zsfdfasga</token_auth_primary_key><token_auth_backup_key>fgff45788787878</token_auth_backup_key><token_auth_secure_paths><token_auth_secure_path>/Video1</token_auth_secure_path><token_auth_secure_path>/Video2</token_auth_secure_path></token_auth_secure_paths></cdn_resource>'

JSON Request example (with internal publishing point):

XML Request example (with external publishing point):

JSON Request example (with external publishing point):
curl -i -X POST -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url
http://onapp.test/cdn_resources.json -d
'{"cdn_resource":{"cdn_hostname":"testLSJ.external",
"resource_type":"STREAM_LIVE",
"publishing_point":"external","publishing_location":"http://www.google.com","failover_publishing_location":"rtmp://test.com/test",
"hotlink_policy":"BLOCK_BY_DEFAULT", "domains":"www.google.com",
"country_access_policy":"ALLOW_BY_DEFAULT", "countries":['AO','BH'], "secure_wowza_on":1, "secure_wowza_token":4t534564tyrt",
"edge_group_ids":['224']}}'}

Where:

- **cdn_hostname** - specify the name which will serve as a label only
- **resource_type** - STREAM_LIVE
- **publishing_point** - the publishing point type: external or internal
- **publishing_location** - specify the URL address for external publishing point. Set the ID of a location that will serve as a publishing point for internal type. For external type, the field can't be blank, must begin with 'rtmp', contain maximum 255 characters and comply with RFC2396.
- **failover_publishing_location** - publishing point failover URL for external publishing point. Specify the ID of a location that will serve as a failover publishing point for internal type. For external type, the field can't be blank, must begin with 'rtmp', contain maximum 255 characters, and comply with RFC2396. The **failover_publishing_location** can't be the same as **publishing_location** parameter.
- **advanced_settings** - set 1 to enable advanced settings
- **hotlink_policy** - configure hotlink policy properties to protect your content from unauthorized hotlinking:
  - **BLOCK_BY_DEFAULT** - block hotlink_policy by default, except for domains specified in the **domains** parameter
  - **NONE** - switch off the rule
- **domains** - domains related to hotlink_policy
- **country_access_policy** - configure a rule to control access to the CDN resource's content for specified countries:
  - **BLOCK_BY_DEFAULT** - block country access policy by default, except for countries specified in the **countries** parameter
  - **NONE** - switch off the country access policy
- **countries** - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.
- **secure_wowza_on** - set 1 to enable secure Wowza streaming encryption, otherwise set 0
- **secure_wowza_token** - specify the Wowza token
- **token_auth_on** - indicate whether Token Authentication is enabled for this resource. It's only supported for VoD PUSH, VoD PULL, and Live Streaming resource.
- **token_auth_primary_key** - set primary key to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt the token at Wowza.
WHITELIST IPS

Server. The key must be alphanumeric with length 6 to 32 characters. This must not be blank if token_auth_on is enabled.

token_auth_backup_key - set backup key (optional) to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt (if primary key failed) the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters.

token_auth_secure_paths - set secure paths that marks streaming to these paths requires a valid token to play a stream. Include only the relative path that appears after the content access point. If undefined, forward slash (/) will be set, means the resource is secured at root level and all streaming request to the resource will be granted only if the provided token is valid. Example - ["/video1", "/video2"]

Response

You will get a response consisting of two parts - the header with HTTP status code and the response body including the parameters. At this stage some of the parameters can be empty. This is expected behavior, because the full process of creation takes some time, and remote service cannot fill in the parameters at this time. For the complete list of parameters use GET request.

Page History

v.3.3.1 adds the following parameters:

- token_auth_on
- token_auth_primary_key
- token_auth.backup_key
- token_auth_secure_paths

27.12 Edit CDN Resource

To edit details of the CDN resource, use the following API call:

PUT /cdn_resources/:id.xml
PUT /cdn_resources/:id.json

XML Request example

curl -i -X PUT -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url 'http://onapp.test/cdn_resources/:id.xml' -d '<cdn_resource><edge_group_ids type="array"><edge_group_id type="integer">1</edge_group_id></edge_group_ids><origin>origin4.com</origin><cdn_hostname>az.test.api</cdn_hostname><cdn_ssl_certificate_id>8</cdn_ssl_certificate_id></cdn_resource>'

JSON Request example
## 27.13 Edit HTTP Pull CDN Resource with Advanced Settings

To edit HTTP Pull CDN resource:

PUT /cdn_resources/:cdn_resource_id.xml
PUT /cdn_resources/:cdn_resource_id.json

**XML Request example**

```
curl -i -X PUT -u user:userpass --url
'Content-type: application/xml' --url
'<cdn_resource><cdn_hostname>testnewnewpull.qwe</cdn_hostname><orig
in>111.111.11.111</origin><cdn_ssl_certificate_id>ssl_sert_id</cdn
_ssl_certificate_id><edge_group_ids type="array"><edge_group_id>225</edge_group_id></edge_group_ids><se
condary_hostnames type="array"><secondary_hostname>test100.com</secondary_hos
name><secondary_hostname>test200.com</secondary_hostname><secondary_hostn
ames><ip_access_policy>BLOCK_BY_DEFAULT</ip_access_policy><ip_addresses>111.11
1.11.111,222.222.22.222</ip_addresses><country_access_policy>ALLOW_BY_DEFAULT</cou
try_access_policy><countries type="array"><country>AL</country><country>GT</country></countries>
<hotlink_policy>ALLOW_BY_DEFAULT</hotlink_policy><domains>abuse.co.ua</domains>
```

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

**Where you can edit all strings:**

- **origin** - the path from which the CDN requests the content (for HTTP Pull CDN resources only). You can specify up to 3 origins.
  - When you are specifying only one origin, it can be either a CDN hostname or an IP address. In case you are specifying more than one origin, they can only be IP addresses.
  - To send two or more origins in the API request, use array. For example:
    ```xml
    <origins type="array">
    <origin>111.111.11.111</origin>
    <origin>111.111.11.111</origin>
    </origins>
    ```

- **cdn_hostname** - indicate the hostname which will serve static content

- **edge_group_ids** - indicate the ID(s) of required CDN edge groups
WHITELIST IPS

- EDIT

WHITELISTED IP

```
abuse.org</domains><url_signing_on>1</url_signing_on><url_signing_key>newurlkey</url_signing_key><cache_expiry>45</cache_expiry><password_on>1</password_on><form_pass><user type="array"><string>user123new</string><string>user234new</string></user><pass type="array"><string>passw123new</string><string>passw234new</string></pass></form_pass><password_unauthorized_html>YOU ARE NOT AUTHORIZED</password_unauthorized_html><flv_pseudo_on>1</flv_pseudo_on><mp4_pseudo_on>1</mp4_pseudo_on><limit_rate>1000</limit_rate><limit_rate_after>1500</limit_rate_after><proxy_cache_key>$host$uri</proxy_cache_key><proxy_read_time_out>60</proxy_read_time_out><proxy_connect_time_out>20</proxy_connect_time_out><http_bot_blocked>1</http_bot_blocked><origin_policy>HTTP</origin_policy></cdn_resource>'
```

JSON Request example

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/cdn_resources/12711.json -d '{}"cdn_resource":{"origin":"test123test.com","cdn_hostname":"jssscdn.apitestpull.com","cdn_ssl_certificate_id":"ssl_cert_id","edge_group_ids":[]","ip_access_policy":"ALLOW_BY_DEFAULT","ip_addresses":"10.10.5.6,125.125.125.125","country_access_policy":"BLOCK_BY_DEFAULT","countries":[]","hotlink_policy":"ALLOW_BY_DEFAULT","domains":"mnw.netggl.com","url_signing_on":1,"url_signing_key":123456789321,"cache_expiry":45,"password_on":1,"form_pass":{"user":["user190","user278"],"pass":["pass123","pass2348"]},"password_unauthorized_html":"<b>You are blocked!</b>"}'}
```

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

Where:
- **origin** - the path from which the CDN requests the content. When using the hostnames according to RFC 1035, the origin may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], and dash [-]. The limit for hostname is 255 characters. You can specify up to 3 origins.
- **cdn_hostname** - indicate the hostname which will serve static content
- **ssl_on** - set to 'true' to enable SSL, otherwise set to 'false'. See the note below for more information.

**SSL**

If the CDN hostname ends with '.r.worldssl.net', SSL will be enabled automatically.
To disable, remove the '.r.worldssl.net' ending and send the "ssl_on":false parameter. To enable, add the '.r.worldssl.net' ending to the cdn_hostname and send the "ssl_on":true parameter.

Be aware that if CDN hostname ends with '.r.worldssl.net', it can not be digit-only (for example 123456.r.worldssl.net is not applicable).

**cdn_ssl_certificate_id** - the ID of the custom SNI SSL certificate you want to add to the resource. You should only specify the IDs of those certificates that were added by the user with whom the new resource will be associated.

**edge_group_ids** - indicate the ID(s) of required CDN edge groups

**secondary_hostnames** - an array of secondary CDN hostnames. You can add up to 7 secondary CDN hostnames.

```
<secondary_hostnames type="array">
  <secondary_hostname>test100.com</secondary_hostname>
  <secondary_hostname>test200.com</secondary_hostname>
</secondary_hostnames>
```

To be able to use a secondary hostname for the CDN resource with SSL enabled, you require an SSL certificate for your custom hostname. For help with questions about the SSL certificate purchase, please contact OnApp support.

**advanced_settings** - set 1 to enable advanced settings:

**ip_access_policy** - configure a rule to control access to the CDN resource's content for a range of IP addresses:

- **ALLOW_BY_DEFAULT** - allow IP access policy by default, except for IP addresses specified in the ip_addresses parameter
- **BLOCK_BY_DEFAULT** - block IP access policy by default, except for IP addresses specified in the ip_addresses parameter
- **NONE** - switch off the IP access policy

**ip_addresses** - IP address(es) related to ip_access_policy parameter; The comma-separated list of IP addresses or IP ranges allowed/blocked by default. Use the following format "10.10.10.10, 20.20.20.0/24"

**hotlink_policy** - configure hotlink policy properties to protect your content from unauthorized hotlinking:

- **ALLOW_BY_DEFAULT** - allow hotlink policy by default, except for domains specified in the domains parameter
- **BLOCK_BY_DEFAULT** - block hotlink_policy by default, except for domains specified in the domains parameter
- **NONE** - switch off the rule
domains - domains related to hotlink_policy

hls_on - set to 1 to enable HTTP Live Streaming (HLS) Optimization

hls_force_cache - set to 1 to create an HTTP rule that will enforce cache expiry. This option is only available when the hls_on parameter is set to 1

country_access_policy - configure a rule to control access to the CDN resource’s content for specified countries:

- ALLOW_BY_DEFAULT - allow country access policy by default, except for countries specified in the countries parameter
- BLOCK_BY_DEFAULT - block country access policy by default, except for countries specified in the countries parameter
- NONE - switch off the country access policy

countries - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.

cache_expiry - set the cache expiry time in minutes (min=1, max=71582788)

url_signing_on - set 1 to enable and protect your files from unauthorized access with a key

url_signing_key - input the key for URL signing. Input letters and digits (6-32 symbols).

password_on - set 1 to enable and to restrict access to the resource (cdn_hostname), otherwise set 0

form_pass - an array with usernames and passwords to access the resource

pass - the user password.

user - the user login, which may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], underscore [ _ ], dash [ - ]. The first symbol should be alphabetic. The username cannot be duplicated.

password_unauthorized_html - text, which will be displayed in case of fail of authentication. Max 1000 chars.

mp4_pseudo_on - set 1 to enable MP4 pseudo streaming, otherwise set 0

flv_pseudo_on - set 1 to enable FLV pseudo streaming, otherwise set 0

ignore_set_cookie_on - set 1 to enable caching content with Set-Cookie response headers, otherwise set 0 to ignore content caching

Nginx Settings

- limit_rate - sets speed limit of a response to a client (per request) in KB/s. Maximum limit rate value - 2147483647 KB/s
- limit_rate_after - sets the amount after which the speed of a response to a client will be limited in KB. Maximum limit rate after value -2147483647 KB
- proxy_read_time_out - proxy server response timeout in seconds. Maximum proxy read timeout value - 65535 seconds
- proxy_connect_time_out - timeout for establishing connection with proxy server in seconds. Maximum proxy connect time out value - 75 seconds.
• **proxy_cache_key** - specify the cache key. You can set the following options:
  o $host$request_uri
  o $host$uri
  o $proxy_host$request_uri
  o $proxy_host$uri

**http_bot_blocked** - set 1 to block Google web crawling bot from indexing the CDN content (for HTTP Pull CDN resources only)

**origin_policy** - set the parameter to choose the type of connection. Possible values are: HTTP, HTTPS, AUTO.

27.13.1 Page history

v 5.1

• added the **hls_force_cache** parameter
• added the **hls_on** parameter

v. 4.0

• Added **cdn_ssl_certificate_id** parameter that enables a user to associate a custom SNI SSL certificate with a CDN resource

v. 3.3.1

Added the following parameter:
• **origin_policy**

27.14 Edit HTTP Push CDN Resource with Advanced Settings

To edit HTTP Push CDN resource:

PUT /cdn_resources/:cdn_resource_id.xml
PUT /cdn_resources/:cdn_resource_id.json

**XML Request example**

curl -i -X PUT -u user:userpass
'<<cdn_resource><cdn_hostname>sfcdn.123apitest.com</cdn_hostname><cdn_ssl_certificate_id>ssl_sert_id</cdn_ssl_certificate_id><edge_grou
WHITELIST IPS

EDIT WHITELISTED IP

```xml
<p_ids type="array"><edge_group_id type="integer">225</edge_group_id><ftp_password>qwertyuiopqwer</ftp_password><secondary_hostnames type="array"><secondary_hostname>namelnewqwertest.com</secondary_hostname><secondary_hostname>namenewqwertest.com</secondary_hostname></secondary_hostnames><ip_access_policy>ALLOW_BY_DEFAULT</ip_access_policy><ip_addresses>111.111.111.111,222.222.222.222</ip_addresses><hotlink_policy>ALLOW_BY_DEFAULT</hotlink_policy><url_signing_on>1</url_signing_on><url_signing_key>newtest12345</url_signing_key><password_on>1</password_on><form_pass><user type="array"><string>user123new</string><string>user234new</string></user><pass type="array"><string>passw123new</string><string>passw234new</string></pass></form_pass><password_unauthorized_html><b>YOU ARE NOT AUTHORIZED</b></password_unauthorized_html><domains>example.com</domains><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><countries>AO</countries><countries>BH</countries><flv_pseudo_on>1</flv_pseudo_on><mp4_pseudo_on>1</mp4_pseudo_on><limit_rate>140</limit_rate><limit_rate_after>11</limit_rate_after></cdn_resource>
```

**JSON Request example**

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/cdn_resources/1234.json -d '{"cdn_resource":{"cdn_hostname":"Jcdn.apitestruhu.com","cdn_ssl_certificate_id":"ssl_cert_id","edge_group_ids": ["225"], "ftp_password": "password123", "secondary_hostnames": ["nameltest12.co", "name2test34.co"], "ip_access_policy": "ALLOW_BY_DEFAULT", "ip_addresses": "10.10.5.6,125.125.125.125", "country_access_policy": "BLOCK_BY_DEFAULT", "countries": ["AL", "AR", "GT", "HR"], "hotlink_policy": "ALLOW_BY_DEFAULT", "domains": "mnw.netggl12.com", "url_signing_on": "1", "url_signing_key": "new123456789321", "password_on": "1", "form_pass": {"user": ["user190", "user278"], "pass": ["pass123", "pass2348"]}, "password_unauthorized_html": "<b>You are blocked!</b>", "mp4_pseudo_on": "1", "flv_pseudo_on": "1", "limit_rate": "80", "limit_rate_after": "13"}}'
```

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

**Where:**

- **origin** - the path from which the CDN requests the content
- **cdn_hostname** - indicate the hostname which will serve static content
- **ssl_on** - set to 'true' to enable SSL, otherwise set to 'false'. See the note below for more information.
### SSL

If the CDN hostname ends with ".r.worldssl.net", SSL will be enabled automatically.

To disable, remove the ".r.worldssl.net" ending and send the "ssl_on":false parameter. To enable, add the ".r.worldssl.net" ending to the cdn_hostname and send the "ssl_on":true parameter.

Be aware that if CDN hostname ends with 'r.worldssl.net', it can not be digit-only (for example 123456.r.worldssl.net is not applicable).

- **cdn_ssl_certificate_id** - the ID of the custom SNI SSL certificate you want to add to the resource. You should only specify the IDs of those certificates that were added by the user with whom the new resource will be associated.

- **edge_group_ids** - indicate the ID(s) of required CDN edge groups

- **ftp_password** - indicate the FTP server password. It should consist of 6-32 alphanumeric symbols.

- **secondary_hostnames** - an array of secondary CDN hostnames. You can add up to 7 secondary CDN hostnames.

```xml
<secondary_hostnames type="array">
  <secondary_hostname>test100.com</secondary_hostname>
  <secondary_hostname>test200.com</secondary_hostname>
</secondary_hostnames>
```

To be able to use a secondary hostname for the CDN resource with SSL enabled, you require an SSL certificate for your custom hostname. For help with questions about the SSL certificate purchase, please contact OnApp support.

- **advanced_settings** - set 1 to enable advanced settings:

  - **ip_access_policy** - configure a rule to control access to the CDN resource's content for a range of IP addresses:
    - **ALLOW_BY_DEFAULT** - allow IP access policy by default, except for IP addresses specified in the ip_addresses parameter
    - **BLOCK_BY_DEFAULT** - block IP access policy by default, except for IP addresses specified in the ip_addresses parameter
    - **NONE** - switch off the IP access policy

  - **ip_addresses** - IP address(es) related to ip_access_policy parameter; the comma-separated list of IP addresses or IP ranges allowedblocked by default. Use the following format "10.10.10.10, 20.20.20.0/24"

  - **hotlink_policy** - configure hotlink policy properties to protect your content from unauthorized hotlinking:
    - **ALLOW_BY_DEFAULT** - allow hotlink policy by default, except for domains specified in the domains parameter
- **BLOCK_BY_DEFAULT** - block hotlink_policy by default, except for domains specified in the `domains` parameter

- **NONE** - switch off the rule

`domains` - domains related to hotlink_policy

`country_access_policy` - configure a rule to control access to the CDN resource's content for specified countries:

- **ALLOW_BY_DEFAULT** - allow country access policy by default, except for countries specified in the `countries` parameter

- **BLOCK_BY_DEFAULT** - block country access policy by default, except for countries specified in the `countries` parameter

- **NONE** - switch off the country access policy

`countries` - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.

`cache_expiry` - set the cache expiry time in minutes

`url_signing_on` - set 1 to enable and protect your files from unauthorized access with a key

`url_signing_key` - input the key for URL signing. Input letters and digits (6-32 symbols).

`password_on` - set 1, if the access to the resource is restricted; otherwise 0

`form_pass` - an array with usernames and passwords to access the resource

`pass` - the user password.

`user` - the user login, which may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], underscore [ _ ], dash [ - ]. The first symbol should be alphabetic. The username cannot be duplicated.

`password_unauthorized_html` – the message that is displayed when there is unauthorized access. Max 1000 chars.

`mp4_pseudo_on` - set 1 to enable MP4 preudo streaming, otherwise set 0

`flv_pseudo_on` - set 1 to enable FLV preudo streaming, otherwise set 0

`ignore_set_cookie_on` - set 1 to enable caching content with Set-Cookie response headers, otherwise set 0

**Nginx Settings**

- **limit_rate** - sets speed limit of a response to a client (per request) in KB/s. Maximum limit rate value - 2147483647 KB/s

- **limit_rate_after** - sets the amount after which the speed of a response to a client will be limited in KB. Maximum limit rate after value - 2147483647 KB
27.14.1 Page History

v. 4.0

- Updated `cdn_hostname` parameter with ability to enable or disable SSL
- Added `cdn_ssl_certificate_id` parameter that enables a user to associate a custom SNI SSL certificate with a CDN resource

27.15 Edit VoD Push CDN Resource with Advanced Settings

To create new video on demand resource with advanced settings:

**PUT /cdn_resources/:cdn_resource_id.xml**

**PUT /cdn_resources/:cdn_resource_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/cdn_resources/:cdn_resource_id.xml -d '<cdn_resource><cdn_hostname>apitest.com</cdn_hostname><ftp_password>testpassword</ftp_password><edge_group_ids type="array"><edge_group_id>466</edge_group_id><edge_group_id>1</edge_group_id></edge_group_ids><hotlink_policy>BLOCK_BY_DEFAULT</hotlink_policy><domains>g.coabuse.ua</domains><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><countries type="array"><country>AL</country><country>GT</country></countries><secure_wowza_on>1</secure_wowza_on><secure_wowza_token>test123456</secure_wowza_token><token_auth_on>1</token_auth_on><token_auth_primary_key>zsfdfasga</token_auth_primary_key><token_auth_backup_key>fgff45788787878</token_auth_backup_key><token_auth_secure_paths type="array"><token_auth_secure_path>/video1</token_auth_secure_path><token_auth_secure_path>/video2</token_auth_secure_path></token_auth_secure_paths></cdn_resource>
```

**JSON Request example**

```
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/cdn_resources/:cdn_resource_id.json -d '{"cdn_resource":{"ftp_password":"testpassword","cdn_hostname":"apitest.com","edge_group_ids":["1"],"hotlink_policy":"BLOCK_BY_DEFAULT","domains":"test.com","country_access_policy":"ALLOW_BY_DEFAULT","countries":["AL","GT"],"secure_wowza_on":"1","secure_wowza_token":"test123456","token_auth_on":"1","token_auth_primary_key":"zsfdfasga","token_auth_secure_paths": ["/video1","/video2"],"token_auth_backup_key":"fgff45788787878"}}'
```
Returns HTTP 204 response on successful processing, and HTTP 404 when there is no CDN resource with a requested ID, or URL is incorrect.

Where:
- **cdn_hostname** - specify the name which will serve as a label only
- **ftp_password** - indicate the FTP server password. It should consist of 6-32 alphanumeric symbols.
- **edge_group_ids** - indicate the ID(s) of required CDN edge groups
- **advanced_settings** - set 1 to enable advanced settings:
  - **hotlink_policy** - configure hotlink policy properties to protect your content from unauthorized hotlinking:
    - **BLOCK_BY_DEFAULT** - block hotlink_policy by default, except for domains specified in the `domains` parameter
    - **NONE** - switch off the rule
  - **country_access_policy** - configure a rule to control access to the CDN resource's content for specified countries:
    - **BLOCK_BY_DEFAULT** - block country access policy by default, except for countries specified in the `countries` parameter
    - **NONE** - switch off the country access policy
- **domains** - domains related to hotlink policy
- **country_access_policy** - configure a rule to control access to the CDN resource's content for specified countries:
  - **BLOCK_BY_DEFAULT** - block country access policy by default, except for countries specified in the `countries` parameter
  - **NONE** - switch off the country access policy
- **countries** - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.
- **secure_wowza_on** - set 1 to enable secure Wowza streaming encryption, otherwise set 0
- **secure_wowza_token** - specify the Wowza token
- **token_auth_on** - indicate whether Token Authentication is enabled for this resource. It's only supported for VoD PUSH, VoD PULL, and Live Streaming resource.
- **token_auth_primary_key** - set primary key to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters. This must not be blank if `token_auth_on` is enabled.
- **token_auth_backup_key** - set backup key (optional) to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt (if primary key failed) the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters.
- **token_auth_secure_paths** - set secure paths that marks streaming to these paths requires a valid token to play a stream. Include only the relative path that appears after the content access point. If undefined, forward slash (/) will be set, means the resource is secured at root level and all streaming request to the resource will be granted only if the provided token is valid. Example - ["/video1", "/video2"]

**Page History**

v.3.3.1 adds the following parameters:
27.16 Edit VoD Pull CDN Resource with Advanced Settings

To create new video on demand resource with advanced settings:

**PUT /cdn_resources/:cdn_resource_id.xml**

**PUT /cdn_resources/:cdn_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/cdn_resources/:cdn_resource_id.xml -d '      <cdn_resource><cdn_hostname>apitest.com</cdn_hostname><origin>111.111.111</origin><edge_group_ids type="array"><edge_group_id>466</edge_group_id><edge_group_id>1</edge_group_id></edge_group_ids><hotlink_policy>BLOCK_BY_DEFAULT</hotlink_policy><domains>goo.coabuse.ua</domains><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><countries type="array"><country>AL</country><country>GT</country></countries><secure_wowza_on>1</secure_wowza_on><secure_wowza_token>test123456</secure_wowza_token><token_auth_on>1</token_auth_on><token_auth_primary_key>zsfdfasga</token_auth_primary_key><token_auth_backup_key>fgff45788787878</token_auth_backup_key><token_auth_secure_paths type="array"><token_auth_secure_path>/video1</token_auth_secure_path><token_auth_secure_path>/video2</token_auth_secure_path></token_auth_secure_paths></cdn_resource>'
```

**JSON Request example**

```bash
curl -i -X PUT -H 'Accept: application/json' -H 'Content-type: application/json' -u user:userpass --url http://onapp.test/cdn_resources/:cdn_resource_id.json -d '{"cdn_resource":{"origin":"test.com","cdn_hostname":"ohcdn235.apitest.com","edge_group_ids": ["1","2"], "hotlink_policy":"BLOCK_BY_DEFAULT","domains": "test.com","country_access_policy": "ALLOW_BY_DEFAULT","countries": ["AL","GT"], "secure_wowza_on": "1", "secure_wowza_token": "test123456","token_auth_on": "1","token_auth_primary_key": "zsfdfasga","token_auth_backup_key": "fgff45788787878","token_auth_secure_paths": ["/video1","/video2"],"token_auth_secure_path": "test123456"}}'
```
WHITELIST IPS - EDIT WHITELISTED IP

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

**Where:**
- *cdn_hostname* - specify the name which will serve as a label only
- *origin* - the path from which the CDN requests the content. When using the hostnames according to RFC 1035, the origin may consist of letters [A-Z a-z] (case insensitive manner), digits [0-9], and dash [-]. The limit for hostname is 255 chars. The VoD Pull resource can have only one origin.
- *edge_group_ids* - indicate the ID(s) of required CDN edge groups
- *advanced_settings* - set 1 to enable advanced settings:
  - *hotlink_policy* - configure hotlink policy properties to protect your content from unauthorized hotlinking:
    - BLOCK_BY_DEFAULT - block hotlink policy by default, except for domains specified in the *domains* parameter
    - NONE - switch off the rule
  - *countries* - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format
- *secure_wowza_on* - set 1 to enable secure Wowza streaming encryption, otherwise set 0
- *secure_wowza_token* - specify the Wowza token
- *token_auth_on* - indicate whether Token Authentication is enabled for this resource. It's only supported for VoD PUSH, VoD PULL, and Live Streaming resource.
- *token_auth_primary_key* - set primary key to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters. This must not be blank if *token_auth_on* is enabled.
- *token_auth_backup_key* - set backup key (optional) to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt (if primary key failed) the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters.
- *token_auth_secure_paths* - set secure paths that marks streaming to these paths requires a valid token to play a stream. Include only the relative path that appears after the content access point. If undefined, forward slash (/) will be set, means the resource is secured at root level and all streaming request to the resource will be granted only if the provided token is valid. Example - ["/video1", "/video2"]
v.3.3.1 adds the following parameters:

- `token_auth_on`
- `token_auth_primary_key`
- `token_auth_backup_key`
- `token_auth_secure_paths`
- `storage_server_location`

### 27.17 Edit Live Streaming CDN Resource with Advanced Settings

PUT `/cdn_resources/:id.xml`
PUT `/cdn_resources/:id.json`

**XML Request example**


`'<cdn_resource><cdn_hostname>apitest.com</cdn_hostname><edge_group_ids type="array"><edge_group_id>1</edge_group_id></edge_group_ids><hotlink_policy>BLOCK_BY_DEFAULT</hotlink_policy><domains>test.test.com</domains><country_access_policy>ALLOW_BY_DEFAULT</country_access_policy><secure_wowza_on>1</secure_wowza_on><secure_wowza_token>test123456</secure_wowza_token><publishing_point>external</publishing_point><publishing_location>rtmp://test-stream.com</publishing_location><failover_publishing_location>rtmp://test-stream2.com</failover_publishing_location><token_auth_on>1</token_auth_on><token_auth_primary_key>zsfdfasga</token_auth_primary_key><token_auth_backup_key>fgff45788787878</token_auth_backup_key><token_auth_secure_paths type="array"><token_auth_secure_path>/Video1</token_auth_secure_path><token_auth_secure_path>/Video2</token_auth_secure_path></token_auth_secure_paths></cdn_resource>'`

**JSON Request example**


"secure_wowza_on": "1", "secure_wowza_token": "test123456", "publishing_point": "external",
"publishing_location": "rtmp://test-stream.com", "failover_publishing_location": "rtmp://test-stream2.com", "token_auth_on": "1", "token_auth_primary_key": "zsfdfasga", "token_auth_secure_paths": ["/video1", "/video2"], "token_auth_backup_key": "fgff45788787878"}'}

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

Where:

- **cdn_hostname** - specify the name which will serve as a label only
- **edge_group_ids** - indicate the ID(s) of required CDN edge groups
- **advanced_settings** - set 1 to enable advanced settings:
  - **hotlink_policy** - configure hotlink policy properties to protect your content from unauthorized hotlinking:
    - **BLOCK_BY_DEFAULT** - block hotlink_policy by default, except for domains specified in the **domains** parameter
    - **NONE** - switch off the rule
  - **country_access_policy** - configure a rule to control access to the CDN resource's content for specified countries:
    - **BLOCK_BY_DEFAULT** - block country access policy by default, except for countries specified in the **countries** parameter
    - **NONE** - switch off the country access policy
  - **domains** - domains related to hotlink policy
  - **countries** - country codes, related to country_access_policy in ISO 3166-1 alpha-2 format.
- **secure_wowza_on** - set 1 to enable secure Wowza streaming encryption, otherwise set 0
- **secure_wowza_token** - specify the Wowza token
- **publishing_point** - the publishing point type: external or internal
- **publishing_location** - specify the URL address for external publishing point. Set the ID of a location that will serve as a publishing point for internal type. For external type, the field can't be blank, must begin with 'rtmp', contain maximum 255 characters and comply with RFC2396.
- **failover_publishing_location** - publishing point failover URL for external publishing point. Specify the ID of a location that will serve as a failover publishing point for internal type. For external type, the field can't be blank, must begin with 'rtmp', contain maximum 255 characters, and comply with RFC2396. The **failover_publishing_location** can't be the same as **publishing_location** parameter.
- **token_auth_on** - indicate whether Token Authentication is enabled for this resource. It's only supported for VoD PUSH, VoD PULL, and Live Streaming resource.
- **token_auth_primary_key** - set primary key to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt the token at Wowza
Server. The key must be alphanumeric with length 6 to 32 characters. This must not be blank if token_auth_on is enabled.

token_auth_backup_key - set backup key (optional) to encrypt Token Authentication security parameters to generate token for streaming playback. This same key is used to decrypt (if primary key failed) the token at Wowza Server. The key must be alphanumeric with length 6 to 32 characters.

token_auth_secure_paths - set secure paths that marks streaming to these paths requires a valid token to play a stream. Include only the relative path that appears after the content access point. If undefined, forward slash (/) will be set, means the resource is secured at root level and all streaming request to the resource will be granted only if the provided token is valid. Example - ["/video1", "/video2"]

Page History
v.3.3.1 adds the following parameters:
- token_auth_on
- token_auth_primary_key
- token_auth_backup_key
- token_auth_secure_paths

27.18 Delete CDN Resource

To delete a CDN resource:
DELETE /cdn_resources/:id.xml
DELETE /cdn_resources/:id.json

XML Request example
  curl -i -X DELETE -u user:userpass http://onapp.test
  /cdn_resources/:id.xml

JSON Request example
  curl -i -X DELETE -u user:userpass
  http://onapp.test/cdn_resources/:id.json

Where you have to specify ID of a CDN resource you want to delete.
Returns HTTP 204 response on successful processing, and HTTP 404 when there is no CDN resource with a requested ID, or URL is incorrect.
27.19 Change CDN Resource FTP Password

Using the following request, you can change FTP password of HTTP Push and VoD PUSH resources:

PUT  /cdn_resources/:cdn_resource_id.xml
PUT  /cdn_resources/:cdn_resource_id.json

XML Request example

JSON Request example

Where:

* `cdn_resource_id` - CDN resource ID
* `ftp_password` - required FTP password

27.20 Prefetch CDN Resource Content

To pre-populate HTTP PULL and PUSH content to the CDN, use the following API call:

POST /cdn_resources/:id/prefetch.xml
POST /cdn_resources/:id/prefetch.json

PLEASE NOTE: You can only prefetch content of HTTP CDN resources.
You can use prefetch CDN resource content API requests with entry slashes:

```xml
<prefetch_paths>/home/123.jpeg</prefetch_paths>
```

as well as without them:

```xml
<prefetch_paths>home/123.jpeg</prefetch_paths>
```

XML Request example

```bash
curl -i -X POST -u user:userpass
http://onapp.test/cdn_resources/:cdn_resource_id/prefetch.xml -d '
<prefetch_paths>/home/123.jpeg</prefetch_paths>' -H
```
WHITELIST IPS - EDIT WHITELISTED IP

JSON Request example
```bash
curl -i -X POST -u user:userpass
http://onapp.test/cdn_resources/:cdn_resource_id/prefetch.json -d
"{"prefetch_paths":="/home/123.jpeg"}" -H 'Accept:application/json'
-H 'Content-type:application/json'
```

Where:

**prefetch_path** *--* path to the file you want to prefetch

To prefetch an array of paths:

**XML Request example**
```bash
curl -i -X POST -u user:userpass
http://onapp.test/cdn_resources/:cdn_resource_id/prefetch.xml -d
```

**JSON Request example**
```bash
curl -i -X POST -u user:userpass
http://onapp.test/cdn_resources/:cdn_resource_id/prefetch.json -d
```

27.21 Purge CDN Resource Content

To remove content from HTTP Pull and PUSH cache, use the following request:
```
POST /cdn_resources/:id/purge.xml
POST /cdn_resources/:id/purge.json
```

**PLEASE NOTE:** You can only purge content of HTTP CDN resources.

```xml
<purge_paths>/home/123.jpeg</purge_paths>
```

You can use purge CDN resource content API requests with entry slashes:
```
< purge_paths > home / 123 . jpeg < / purge_paths >
```

**XML Request example**
```bash
curl -i -X POST -u user:userpass
```

**POST /cdn_resources/:id/purge.xml**
**POST /cdn_resources/:id/purge.json**
JSON Request example

curl -i -X POST -u user:userpass
http://onapp.test/cdn_resources/:cdn_resource_id/purge.xml -d
' -H 'Accept: application/xml' -H 'Content-type: application/xml'

Where:

*purge_path* - path to the content you want to remove

To purge an array of paths:

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/cdn_resources/:id/purge.xml -d
' -H 'Accept: application/xml' -H 'Content-type: application/xml'

Where you have to specify an array of paths to the content you want to remove.

To purge all content:

XML Request example

curl -i -u user:userpass -X POST
http://onapp.test/cdn_resources/:cdn_resource_id/purge_all.xml -H
'Accept: application/xml' -H 'Content-type: application/xml'

JSON Request example

curl -i -u user:userpass -X POST
http://onapp.test/cdn_resources/:cdn_resource_id/purge_all.json -H
'Accept: application/json' -H 'Content-type: application/json'

Where you have to specify a CDN resource ID.

If you are using version 3.0 or earlier, use the following request:

XML Request example

curl -i -X POST -u user:userpass
http://onapp.test/cdn_resources/:id/purge.xml -d
' -H 'Accept: application/xml' -H 'Content-type: application/xml'
27.22 View CDN Resource Bandwidth Statistics

To get bandwidth statistics for the resources or a particular resource, use the following request:

GET /cdn_resources/bandwidth.xml
GET /cdn_resources/bandwidth.json

Bandwidth statistics is returned in gigabytes in the SI format (1 GB = 1000 MB). You can also define a shorter period, specify a particular resource or location, set the type and specify how to sort the result returned:

XML Request example
```
curl -i -X GET -u user:userpass
"http://onapp.test/cdn_resources/bandwidth.xml" -d
```

JSON Request example
```
curl -i -X GET -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' --url
"http://onapp.test/cdn_resources/bandwidth.json" -d
'"{"bandwidth":{"start":"2014-11-25T18:25","end":"2014-11-27T11:43","resources":[977655738,389478438,725618714],"locations":[532,331],"type":"GB","group_by":"location"}}'
```

Where:
```
start - the start date to generate statistics in the YYYY-MM-DD+hh:mm:ss format
```
end – the end date to generate statistics in the YYYY-MM-DD+hh:mm:ss format
resources – the identifier of the resource in Aflexi database. To get the identifier, check with cdn_reference parameter in the GET /cdn_resources/:id.{format} request
locations – the ID of the location
type – the statistics type (MBPS or GB). In MBPS mode you can get statistics for the last 10 days only. The older statistics is removed. There are no restrictions for GB mode.
group_by - to get the bandwidth statistics breaking down per location or per resource via API, use the group_by parameter with two possible values: location and resource. In case group_by = location bandwidth stats is breaking down per location, and if group_by = resource is breaking down per resource. This is the optional parameter.
XML output example with resources grouped by resource:

Status: 200 OK
Connection: close
Transfer-Encoding: chunked
Content-Type: application/xml; charset=utf-8

<?xml version="1.0" encoding="UTF-8"?>
<stats type="array">
  <stat>
    <date type="datetime">2014-11-26T18:25:00+02:00</date>
    <resources type="array">
      <resource>
        <977655738>
          <cached type="float">1.4193601249999999</cached>
          <non_cached type="float">0.0</non_cached>
        </977655738>
      </resource>
      <resource>
        <389478438>
          <cached type="float">1.4193601249999999</cached>
          <non_cached type="float">0.0</non_cached>
        </389478438>
      </resource>
    </resources>
  </stat>
  <stat>
    <date type="datetime">2014-11-25T18:25:00+02:00</date>
    <resources type="array">
      <resource>
        <977655738>
          <cached type="float">0.425808015</cached>
          <non_cached type="float">0.047311991</non_cached>
        </977655738>
      </resource>
      <resource>
        <725618714>
        </resource>
    </resources>
  </stat>
</stats>
XML request example with resources without group_by parameter:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<stats type="array">
  <stat>
    <date type="datetime">2014-11-25T18:25:00+02:00</date>
    <cached type="float">2.0390744709999997</cached>
    <non_cached type="float">0.22656379899999998</non_cached>
  </stat>
  <stat>
    <date type="datetime">2014-11-26T18:25:00+02:00</date>
    <cached type="float">2.8387202499999997</cached>
    <non_cached type="float">0.0</non_cached>
  </stat>
</stats>
```

Where:
- **non_cached** – the amount of content which is not cached
- **cached** – the amount of data cached
- **date** – the point of time for which the statistics is generated

The frequency of the points of time for which the statistics is generated depends on the period of time of requested statistics:

<table>
<thead>
<tr>
<th>Time Period</th>
<th>Frequency in seconds</th>
<th>Equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 30 days</td>
<td>86400</td>
<td>1 day</td>
</tr>
<tr>
<td>31 - 93 days</td>
<td>604800</td>
<td>1 week</td>
</tr>
<tr>
<td>equal or more than 93 days</td>
<td>1209600</td>
<td>2 weeks</td>
</tr>
</tbody>
</table>

Statistics available in the frequency higher than selected will be accumulated to a single point of applicable frequency.

E.g. The statistics was requested for the period of 31-93 days, so the frequency of points in the graph is 7 days. If the statistics was generated few times during those 7 days (day1+day2+dayn) it will be added up and displayed as a single point, with a time stamp
marked as the first day of such 7 days.

27.23 View CDN Resource Streaming Statistics

To view streaming statistics for the live streaming CDN resources or a particular resource, use the following request:

GET /cdn_resources/stream_stats.xml
GET /cdn_resources/stream_stats.json

You can also define a shorter period, specify a particular resource or location and set the type:

GET /cdn_resources/stream_stats.xml?start=2011-09-01&end=2012-09-01&resources[]=787341593&locations[]=18
GET /cdn_resources/stream_stats.json?start=2011-09-01&end=2012-09-01&resources[]=787341593&locations[]=18

PLEASE NOTE: In Bash shell you have to use backslashes to enclose square brackets (see example below):

```
resources[]=787341593&locations[]=18
```

Where:

- **start** – the start date to generate statistics in the YYYY-MM-DD format
- **end** – the end date to generate statistics in the YYYY-MM-DD format
- **resources** – the identifier of the resource in Aflexi database. To get the identifier, check with cdn_reference parameter in the GET /cdn_resources/:id.{format} request
- **locations** – the ID of the location

27.24 View CDN Resource Billing Statistics

To view billing statistics for a resource:

GET /cdn_resources/:id/billing.xml
GET /cdn_resources/:id/billing.json

You can also define a shorter period by setting Start and End date (set use_local_time to 1 to use local time):

GET /cdn_resources/:id/billing.xml?period[startdate]=YYYY-MM-
OnApp 5.3 API Guide  v2

XML Output example:
HTTP/1.1 200 OK
Date: Thu, 01 Nov 2012 08:46:57 GMT
Server: Apache/2.2.3 (CentOS)
X-UA-Compatiible: IE=Edge,chrome=1
ETag: "5639dcb4af97cc6b5614f0e7102414a6"
Cache-Control: must-revalidate, private, max-age=0
X-Request-Id: b4aa84f305441fa8e791a2b597ed0c8e
X-Runime: 0.064716
X-Rack-Cache: miss
Set-Cookie: _session_id=e01cb0aac4fc66becbd5bb294966d12e; path=/; HttpOnly
Status: 200
Connection: close
Transfer-Encoding: chunked
Content-Type: application/xml; charset=utf-8
<?xml version="1.0" encoding="UTF-8"?>
<user_hourly_stats type="array">
  <user_hourly_stat>
    <cost type="float">1.51980002556229e-05</cost>
    <edge_group_id type="integer">78</edge_group_id>
    <edge_group_label>OH_203</edge_group_label>
    <stat_time type="datetime">2012-10-10T14:00:00+03:00</stat_time>
    <value type="decimal">0.0</value>
  </user_hourly_stat>
</user_hourly_stats>

Where:
cost - the total due for this resource
dge_group_id - the ID of the edge group
dge_group_label - the label of the edge group
stat_time - time when the statistics was gathered
value - traffic value

27.25 View CDN Resource Raw Log Configuration

To view the raw log configuration, use the following request:
GET /cdn_resources/raw_log.xml
GET /cdn_resources/raw_log.json

XML Request Example:
curl -i -X GET -u user:userpass -H 'Accept: application/xml' -H
'Content-type: application/xml' --url
"http://onapp.test/cdn_resources/raw_log.xml"

JSON Request Example:
curl -i -X GET -u user:userpass -H 'Accept: application/json' -H
'Content-type: application/json' --url
"http://onapp.test/cdn_resources/raw_log.json"

XML Response Example:
Disabled:
<raw_log>
  <protocol></protocol>
</raw_log>

Where:
protocol - delivery protocol

For the FTP/SFTP delivery protocol:
<raw_log>
  <protocol>ftp</protocol>
  <uri>rawlog.com</uri>
  <user>username</user>
  <pass>password</pass>
</raw_log>

Where:
protocol - delivery protocol
uri - the hostname of the server to which the log will be delivered
user - the user name of the FTP/SFTP client on the server to which the log will be delivered
pass - the password of the FTP/SFTP client on the server to which the log will be delivered

For the Syslog delivery protocol:
<raw_log>
  <protocol>syslog</protocol>
  <uri>rawlog.com</uri>
  <syslog_protocol>tcp</syslog_protocol>
  <port>80</port>
</raw_log>

Where:
protocol - delivery protocol
uri - the hostname of the server to which the log will be delivered
syslog_protocol - the protocol that will be used for sending the log: TCP or UDP
port - the port number of the syslog the server to which the log will be delivered
27.26 Edit CDN Resource Raw Log Configuration

To edit the raw log configuration, use the following request:

POST /cdn_resources/raw_log.json
POST /cdn_resources/raw_log.json

To disable raw logs:

XML Request Example:

```
```

JSON Request Example:

```
```

Where:

- **protocol** - delivery protocol
- **uri** - the hostname of the server to which the log will be delivered
- **user** - the user name of the FTP/SFTP client on the server to which the log will be delivered
- **pass** - the password of the FTP/SFTP client on the server to which the log will be delivered

To configure the FTP/SFTP delivery protocol:

XML Request Example:

```
```

JSON Request Example:

```
```

Where:

- **protocol** - delivery protocol
- **uri** - the hostname of the server to which the log will be delivered
- **user** - the user name of the FTP/SFTP client on the server to which the log will be delivered
- **pass** - the password of the FTP/SFTP client on the server to which the log will be delivered

To configure the Syslog delivery protocol:
OnApp 5.3 API Guide  v2

WHITELIST IPS

EDIT WHITELISTED IP

379
<id type="integer">11639</id>
<resource_type><resource_type>STREAM_LIVE</resource_type>
<updated_at>2013-08-06T16:39:30+03:00</updated_at>
<user_id type="integer">20</user_id>
<last_24h_cost type="float">0.0</last_24h_cost>
<cdn_reference type="integer">881661104</cdn_reference>
<publishing_point>internal</publishing_point>

<instructions>
<streaming>
<credentials>
<username>P881661104</username>
<password>bUHzhj61MG</password>
</credentials>
<urls>
<fms>rtmp://881661104.publishstream.worldcdn-beta.net/P881661104</fms>
<backup>rtmp://backup.881661104.publishstream.worldcdn-beta.net/P881661104</backup>
</urls>
</streaming>
<publishing>
<urls>
<smil>http://video.worldcdn-beta.net/881661104/mystream.smil</smil>
<apple_http_live_streaming>http://video.worldcdn-beta.net/881661104/_definst_/mystream.m3u8</apple_http_live_streaming>
<adobe_http_dynamic_streaming>http://video.worldcdn-beta.net/881661104/_definst_/mystream.f4m</adobe_http_dynamic_streaming>
<microsoft_smooth_streaming>http://video.worldcdn-beta.net/881661104/_definst_/mystream.ism</microsoft_smooth_streaming>
</urls>
</publishing>
</instructions>
</cdn_resource>

Where:
* cdn_hostname* - indicate the hostname which will serve the content
  * cname - a CNAME for the CDN Hostname which can then be used to view the contents.
  * resource_type - the Stream Live resource
  * id - the ID of the resource
  * user_id - the user ID
  * last_24h_cost - cost of the resource for the last 24 hours.
  * cdn_reference - the ID in OnApp Dashboard
  * publishing_point - the publishing point type: internal
instructions - the array of parameters for embedding video and enabling live streaming for Live Streaming CDN resources.
credentials - the array with user credentials:
username - the user log in
password - the user password
stream - the stream name
urls - the array with URLs
fms - the FMS URL
backup - the Backup URL
smil - The SMIL playlist provides an RTMP URL and should be used with Flash-based players only.
apple_http_live_streaming - This returns a 302 redirect to a Apple HLS manifest and should be used with Apple HLS-compatible players only.
adobe_http_dynamic_streaming - This returns an Adobe HDS manifest and should be used with Adobe HDS-compatible players only.
microsoft_smooth_streaming - This returns a 302 redirect to Smooth Streaming manifest and should be used with Smooth Streaming-compatible players only.

27.28 Search CDN Resource

To search for a specific CDN resource, use the following request:
GET /cdn_resources.xml?q=search_param
GET /cdn_resources.json?q=search_param
The request will search the CDN hostname, origin (both the hostname and an IP address, including redundant origins). For Live Stream CDNs, the request searches the pattern inside Publishing Location fields (main and failover) and Aflexi ID.
XML Request example

JSON Request example

The example request will search all the CDN resources where the 111.111.111.1 IP is used.
XML output example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<cdn_resources type="array">
  <cdn_resource>
    <cdn_hostname>oldtestnewnewpull.qwe</cdn_hostname>
    <cname>438335686.r.worldcdn-beta.net</cname>
    <created_at type="datetime">2013-07-</cdn_resource>
</cdn_resources>
```
WHITELIST IPS - EDIT WHITELISTED IP

Where:
cdn_hostname - the hostname which will serve static content
cname - CNAME record
created_at - the date when the resource was created
id - the resource ID in the database
resource_type - HTTP PULL or PUSH
updated_at - the date when the resource was updated
user_id - the ID of the user, who owns the resource
last_24h_cost - the amount of money owed for the resource for the last 24 hours.
cdn_reference - the identifier in database
origins - the path from which the CDN requests the content

27.29 Suspend CDN Resource

To suspend a specific CDN resource, use the following request:
PUT /cdn_resources/resource_id/suspend.xml
PUT /cdn_resources/resource_id/suspend.json

XML Request example:
curl -i -X PUT -u user:userpass --url

JSON Request example:
curl -i -X PUT -u user:userpass --url
Where:
resource_id - the ID of the CDN resource which you wish to suspend.

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

### 27.30 Resume CDN Resource

To resume a specific suspended CDN resource, use the following request:

PUT /cdn_resources/resource_id/resume.xml
PUT /cdn_resources/resource_id/resume.json

**XML Request example:**


**JSON Request example:**


Where:
resource_id - the ID of the CDN resource which you wish to resume.

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

### 27.31 View CDN Advanced Reporting

To view CDN advanced reporting for HTTP PULL and HTTP PUSH resources:
GET /cdn_resources/resource_id/advanced_reporting.xml
GET /cdn_resources/resource_id/advanced_reporting.json

Get Advanced Bandwidth Reporting (including Cache utilization) XML request example:

curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources/resource_id/advanced_reporting.xml
-d '<report><start>YYYY-MM-DD HH:MM:SS</start><end>YYYY-MM-DD HH:MM:SS</end>
type="array"><location>location_id</location></locations></report>'
-H 'Content-type: application/xml'

Get Advanced Bandwidth Reporting (including Cache utilization) JSON Request example:

curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources/resource_id/advanced_reporting.json

Where:
report - the array with the time points for generating statistics
start - the start date of the period for which the statistics should be generated
end - the end date of the period for which the statistics should be generated
locations - the location ID for which the statistics should be generated

The default request returns Bandwidth and Caching report for the last five days covering all Locations.

Default XML Request:
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources/resource_id/advanced_reporting.xml
-H 'Accept: application/xml' -H 'Content-type: application/xml'

Get Advanced Status Codes Reporting XML API request example:
(for HTTP Pull request only)
curl -i -X GET -u user:userpass
http://onapp.test/cdn_resources/resource_id/advanced_reporting.xml
-d '<report><stats_type>status_codes</stats_type><start>YYYY-MM-DD HH:MM:SS</start><end>YYYY-MM-DD HH:MM:SS</end>
type="array"><location>location_id</location></locations></report>'
-H 'Content-type: application/xml'

Get Advanced Status Codes Reporting Json API request example:
(for HTTP Pull request only)

```bash
curl -i -X GET -u user:userpass 
http://onapp.test/cdn_resources/resource_id/advanced_reporting.json 
```

Where:
- `report` - the array with the time points for generating statistics
- `stats_type` - required parameter, in this case `status_codes`
- `start` - the start date of the period for which the statistics should be generated
- `end` - the end date of the period for which the statistics should be generated
- `locations` - the location ID for which the statistics should be generated

### 27.32 Get List of Available Storage Locations

To create a CDN PUSH resource, it is necessary to specify a storage location. To view the list of available storage locations, run the following request:

GET /cdn_resources/available_storage_server_locations.xml
GET /cdn_resources/available_storage_server_locations.json

**XML Request Example**

```bash
curl -i -X GET -H 'Accept: application/xml' -H 'Content-type: application/xml' -u user:password
'http://onapp.test/cdn_resources/available_storage_server_locations.xml?type=streaming&only_active=false'
```

**Json Request Example**

```bash
curl -i -X GET -H 'Accept: application/json' -H 'Content-type: application/json' -u user:password
'http://onapp.test/cdn_resources/available_storage_server_locations.json?type=streaming&only_active=false'
```
Where

* `type` - the optional parameter which describes the Storage Server's type - streaming or http;
* `only_active` - the optional parameter which describes the Storage Server's status (only_active = true if only all active SSs or only_active = false if all the available SSs).

On success HTTP 200 status is returned.

**XML Response Example:**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<storage_server_locations type="array">
  <storage_server_location>
    <city>Kolin</city>
    <country>CZ</country>
    <id type="integer">816382921</id>
    <storage_server>
      <id type="integer">670906719</id>
      <status>ACTIVE</status>
      <http_push_on type="boolean">false</http_push_on>
      <vod_push_on type="boolean">true</vod_push_on>
    </storage_server>
  </storage_server_location>
</storage_server_locations>
```

Where:

* `city` - the city where the storage server is located
* `country` - the country where the storage server is located
* `id` - the ID of the location
* `storage_server` - the array of the storage server details
* `id` - the ID of the storage server
* `status` - the status of the storage server
* `http_push_on`
* `vod_push_on`
28 CDN SSL CERTIFICATES

OnApp customers can import their own SSL certificates with the Subject Name Indication (SNI) extension. One SSL certificate can be associated with several CDN resources, but a resource can only be linked to one SSL certificate.

However, some of the older browsers do not support SNI. In this case, users who prefer browsers that do not support SNI can purchase an SSL certificate and the SAN solution will be applied. On questions about the SSL certificate purchase, please contact OnApp support.

For the list of browsers that do not support SNI, kindly refer to the Server Name Indication article.

OnApp currently supports the following types of certificates:

- domain-valid (DV) certificate (example.com)
  - single certificate
  - wildcard certificate (*.example.com)
  - SAN certificate (any domains)

- organization validation (OV) certificates
  - single certificate
  - wildcard certificate (*.example.com)
  - SAN certificate (any domains)

- extended validation (EV) certificates
  - single certificate
  - wildcard certificate (*.example.com)
  - SAN certificate (any domains)

- high-assurance certificates

- This feature is available for HTTP Pull and HTTP Push resources only.

- To add custom SNI SSL certificates, the user needs to have CDN resources in the cloud and CDN SSL Certificates permissions.

- Custom SNI SSL certificates can be used for secondary hostnames.

- A custom SNI SSL certificate can only be associated with a CDN resource if the certificate and the resource have the same owner. When you link a resource to a certificate you should only specify the IDs of those certificates that were added by the user with whom the new resource will be associated. If you indicate some other certificate’s ID an error will occur.
When a custom SNI SSL certificate is associated with a CDN resource, the certificate applies only to the edge servers subscribed to that resource.

### 28.1 Get List of Custom SNI SSL Certificates

To view the list of available custom SNI SSL certificates, use the following request:

GET /cdn_ssl_certificates.xml
GET /cdn_ssl_certificates.json

**XML Request example:**
```
```

**JSON Request example:**
```
```

**XML Output example:**
```
<?xml version="1.0" encoding="UTF-8"?>
<cdn_ssl_certificates type="array">
  <cdn_ssl_certificate>
    <created_at type="datetime">2015-03-25T14:20:26+00:00</created_at>
    <id type="integer">1</id>
    <name>cert3</name>
    <updated_at type="datetime">2015-03-25T14:20:26+00:00</updated_at>
    <user_id type="integer">3</user_id>
  </cdn_ssl_certificate>
  ...
</cdn_ssl_certificates>
```

**Where:**
- `created_at` - the time in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- `id` - ID of the certificate
- `name` - name of the certificate
- `updated_at` - the time in the [YYYY][MM][DD]T[hh][mm][ss]Z format
- `user_id` - the ID of the user who added the certificate
- `cdn_reference` - the ID of the certificate on the remote Aflexi server
28.2 Get Custom SNI SSL Certificate Details

To view the details of a custom SNI SSL certificate, use the following request:

GET /cdn_ssl_certificates/:id.xml
GET /cdn_ssl_certificates/:id.json

XML Request example:
```
```

JSON Request example:
```
```

Where:
- `id` - ID of the certificate

XML Output example:
```
<?xml version="1.0" encoding="UTF-8"?>
<cdn_ssl_certificate>
  <created_at type="datetime">2015-04-08T03:52:00-10:00</created_at>
  <id type="integer">35</id>
  <name></name>
  <updated_at type="datetime">2015-04-08T03:52:00-10:00</updated_at>
  <user_id type="integer">30</user_id>
  <cdn_resources type="array">
    <cdn_resource>
      <cdn_hostname>res1.test.com</cdn_hostname>
      <cdn_ssl_certificate_id type="integer">35</cdn_ssl_certificate_id>
      <cname>535478274.r.worldcdn-beta.net</cname>
      <created_at type="datetime">2014-11-25T23:29:54-10:00</created_at>
      <id type="integer">7401</id>
      <resource_type>HTTP_PULL</resource_type>
      <updated_at type="datetime">2015-04-09T04:30:29-10:00</updated_at>
      <user_id type="integer">30</user_id>
      <last_24h_cost type="float">0.0</last_24h_cost>
      <cname>535478274.r.worldcdn-beta.net</cname>
      <cdn_reference type="integer">535478274</cdn_reference>
      <origins type="array">
        <origin>1.1.1.1</origin>
```

```
<cdn_resource>
<cdn_reference type="integer">729656986</cdn_reference>
</cdn_resource>

<cdn_ssl_certificate>

Where:
created_at - the time in the [YYYY][MM][DD][hh][mm][ss]Z format
id - ID of the certificate
name - name of the certificate
updated_at - the time in the [YYYY][MM][DD][hh][mm][ss]Z format
user_id - ID of the user who added the certificate

cdn_resources - array of parameters related to the CDN resources associated with
the certificate:
cdn_hostname - the hostname that will serve static content
cdn_ssl_certificate_id - the ID of the custom SNI SSL certificate associated with
the resource
cname - CNAME record
created_at - the date when the CDN resource was created in the
[YYYY][MM][DD][hh][mm][ss]Z format
id - the resource ID in the database
resource_type - CDN resource type
updated_at - the date when the CDN resource was updated in the
[YYYY][MM][DD][hh][mm][ss]Z format
user_id - the ID of the user, who owns the resource
last_24h_cost - the amount due for the last 24 hours
cdn_reference - ID of the SSL certificate on the remote Aflexi server
origin - the path from which the CDN requests the content
cdn_reference - ID of the certificate on the remote Aflexi server

28.3 Add Custom SNI SSL Certificate

To add a custom SNI SSL certificate, use the following request:
POST /cdn_ssl_certificates.xml
POST /cdn_ssl_certificates.json

To add custom SNI SSL certificates, the user needs to have CDN resources in the cloud
and CDN SSL Certificates permissions.

XML Request example

curl -i -X POST -u user:userpass -H 'Accept: application/xml' -H 'Content-type: application/xml' --url 'http://onapp.test/cdn_ssl_certificates.xml' -d '<cdn_ssl_certificate><name>cert1</name><cert>-----BEGIN CERTIFICATE-----
\r\n\nMIIFKzCCBBoGAwIBAgIQMvEFlcrw7X8kOaJ/SyleYjANBgkqhkiG9w0BAQUFADB
\nMQswCQYDVQQGEwJCRTEfMB0GA1UECxMWRm9yIFRlc3QgUHVycG9zZXMgT25seT
EZ\r\n\nMBcGA1UEChMQR2xvYmFsU21nbipIzYTE0MDIGA1UEAxMrR2xvYmFsU21nb...
OnApp 5.3 API Guide

WHITELIST IPS

JSON Request example

curl -i -X POST -u user:userpass -H 'Accept: application/json' -H 'Content-type: application/json' "http://onapp.test/cdn_ssl_certificates.json" -d '{"cdn_ssl_certificate":{"name":"cert1","cert":"-----BEGIN CERTIFICATE-----
MIIFKzCCBiOgAwIBAgIQMvEFlcrw7X8kOaJ/Sy1eYjANBgkqhkiG9w0BAQUFADB/
MIIBIjANBgkqhkiG9w0BAQUFADB/MIICvQIBADANBgkqhkiG9w0BAQUFADB/MIIDbA
-----END CERTIFICATE-----","key":"-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAuFmvC2aa7w6OnHlFS6jujug6BURiALkXmM7QpZGpnpAkmrs
M
-----END RSA PRIVATE KEY-----"}'

WHITELISTED IP
WHITELIST IPS

EDIT WHITELISTED IP