Upgrade Guide for Cloud with CloudBooted Servers

This guide presents the complete walk-through how to upgrade OnApp Cloud v4.2 to the v4.3 for the cloud configuration where all servers are CloudBooted except Control Panel server. Please follow the complete procedure of the upgrade process. All packages (Control Panel, CloudBoot, Compute resources) must belong to the same major version to ensure the best performance of your cloud.

See also:

Installation Guide

Technical Details

On this page:

Important Notes

1. You must be running the latest patch of OnApp 4.2 version to upgrade to 4.3 version. If you are using an earlier version, please upgrade to 4.2. first.
2. Check the Activity Log in your OnApp CP dashboard if there are no transactions running in your cloud. If so, wait until all transactions are complete.
3. Make sure no Control Panel files are open for editing under the root user account.
4. If you are using a third-party billing platform, please ensure that this is compatible with OnApp 4.3 before proceeding with the upgrade! The latest WHMCS modules can be found here.
5. If you are using WHMCS modules, make sure to update the PHP Wrapper after you update OnApp Cloud. Download the latest wrapper.
6. We strongly recommend that you test all your custom scripts before upgrading your production environment.
7. To avoid VNC console issues, make sure that *ServerName* Apache settings match the SSL certificate.
8. Note that if you are using Floating IPs in your environment or if you have VS with primary IPs which could respond to your Control Panel server from elsewhere on your network we would recommend to disable the new '*Ping hosted virtual servers before initiating failover*' setting to avoid the possibility of a false-positive ICMP result.
9. Be aware that OnApp does not support UEFI on static compute resources. You should disable UEFI on your compute resources before installing OnApp.
10. To provide more stable daemon workflow, for deployments with a high load we recommend increasing values for the following parameters in the */onapp/interface/config/on_app.yml* file:

Parameter	Recommended value
backup_taker_delay	300
schedule_runner_delay	60
transaction_runner_delay	300

- Drives assigned for use by Integrated Storage are identified using a disk signature that is generated using SCSI page query mechanism to the device. Please note that disk signatures may change across different kernel versions following an upgrade and reboot. If this occurs, go to the compute resource edit page to re-identify and select the correct drives. Please contact support if you have any concerns regarding this operation.

Upgrade Control Panel Server

- CP installer for Installation and Upgrade contains a new *-D* option enabling to avoid OnApp database dumping during the install/upgrade.
- To increase the cloud performance we recommend setting RUBY_GC_MALLOC_LIMIT parameter in custom configurations to 16 millions. For more information on RUBY_GC_MALLOC_LIMIT parameter, refer to Ruby's GC Configuration and Garbage Collection articles.
- Installer output is redirected to ./onapp-cp-install.log
- All installer critical errors are in /var/log/messages
- Custom values must be set before the installer script runs.
- You may wish to reboot your Control Panel server to take advantage of a new kernel if it is installed. It is not required immediately as a part of the upgrade process though.

To upgrade your Control Panel server:

1. Stop the Monit service:

```
# service monit stop
```

2. Run the following command from the CP server to stop the OnApp service:

```
service onapp stop
```

3. Download and install the latest OnApp YUM repository file:

```
bash#> rpm -Uvh http://rpm.repo.onapp.com/repo/onapp-repo-4.3.noarch.rpm
```

4. Upgrade OnApp Control Panel installer package:

```
bash#> yum update onapp-cp-install
```

5. Update your server OS components (if required):

```
bash# /onapp/onapp-cp-install/onapp-cp-install.sh -y
```

6. *(Optional)* If you need some custom Control Panel configuration, set the values before the installer script runs.

Template server URL

```
TEMPLATE_SERVER_URL='http://templates-manager.onapp.com';
```

# IPs (separated with coma) list for the snmp to trap

```
SNMP_TRAP_IPS=
```

# OnApp Control Panel custom version

```
ONAPP_VERSION=""
```

# OnApp MySQL/MariaDB connection data (database.yml)

```
ONAPP_CONN_WAIT_TIMEOUT=15
ONAPP_CONN_POOL=30
ONAPP_CONN_RECONNECT='true'
ONAPP_CONN_ENCODING='utf8'
ONAPP_CONN_SOCKET='/var/lib/mysql/mysql.sock'
```

# MySQL/MariaDB server configuration data (in case of local server)

```
MYSQL_WAIT_TIMEOUT=604800
MYSQL_MAX_CONNECTIONS=500
MYSQL_PORT=3306
```

# Use MariaDB instead of MySQL as OnApp database server (Deprecated parameter. If you set any values for this parameter, they will not take effect)

```
WITH_MARIADB=0
```

# Configure the database server relative amount of available RAM

```
TUNE_DB_SERVER=1
```

# The number of C data structures that can be allocated before triggering the garbage collector. It defaults to 8 million

```
RUBY_GC_MALLOC_LIMIT=16000000
```

# sysctl.conf net.core.somaxconn value

```
NET_CORE_SOMAXCONN=2048
```

# The root of OnApp database dump directory (on the Control Panel box)

```
ONAPP_DB_DUMP_ROOT=""
```

# Remote server's (to store database dumps) IP, user, path, openssh connection options ans number of dumps to keep

```
DB_DUMP_SERVER=""
DB_DUMP_USER="root"
DB_DUMP_SERVER_ROOT="/onapp/backups"
DB_DUMP_SERVER_SSH_OPT="-o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null -o
PasswordAuthentication=no"
KEEP_DUMPS=168
DB_DUMP_CRON='40 * * * *'
```

# Enable monit - tool for managing and monitoring Unix systems

```
ENABLE_MONIT=1
```

# If enabled (the 1 value is set) - install (if local box) and configures RabbitMQ Server (messaging system) for the vCloud support. (Deprecated parameter. If you set any values for this parameter, they will not take effect)

```
ENABLE_RABBITMQ=1
```

# Rotate transactions' log files created more than TRANS_LOGS_ROTATE_TIME day(s) ago

```
TRANS_LOGS_ROTATE_TIME=30
```

# Maximum allowed for uploading file size in bytes, from 0 (meaning unlimited) to 2147483647 (2GB). Default is 1GB

```
MAX_UPLOAD_SIZE=1073741824
```

# Timeout before ping Redis Server to check if it is started. Default is 5 sec.

```
REDIS_PING_TIMEOUT=5
```

# OnApp Control Panel SSL certificates (please do not change if you aren't familar with SSL certificates)
# * The data below to generate self-signed PEM-encoded X.509 certificate

```
SSL_CERT_COUNTRY_NAME=UK
SSL_CERT_ORGANIZATION_NAME='OnApp Limited'
SSL_CERT_ORGANIZATION_ALUNITNAME='OnApp Cloud'
SSL_CERT_COMMON_NAME=`hostname --fqdn 2>/dev/null`
```

#   SSLCertificateFile, SSLCertificateKeyFile Apache directives' values
#   ssl_certificate, ssl_certificate_key Nginx directives' values

```
SSLCERTIFICATEFILE=/etc/pki/tls/certs/ca.crt
SSLCERTIFICATECSRFILE=/etc/pki/tls/private/ca.csr
SSLCERTIFICATEKEYFILE=/etc/pki/tls/private/ca.key
```

# * PEM-encoded CA Certificate (if custom one exists)
#   SSLCACertificateFile, SSLCertificateChainFile Apache directives' values
#   ssl_client_certificate Nginx directives' values

```
SSLCACERTIFICATEFILE=
SSLCERTIFICATECHAINFILE=
```

#   SSLCipherSuite, SSLProtocol Apache directives' values
#   ssl_ciphers, ssl_protocols Nginx directives' values

```
SSLCIPHERSUITE=
SSLPROTOCOL=
```

```
bash# vi /onapp/onapp-cp.conf
```

> ⚠ If the `onapp-cp.conf` file is not configured correctly, it will replace the SSL files with a self-signed even if a legitimate certificate is already installed.

7. Run Control Panel installer:

```
bash#> /onapp/onapp-cp-install/onapp-cp-install.sh
```

⚠ Customers running vCloud integration, run Control Panel installer with specified rake task:

```
# /onapp/onapp-cp-install/onapp-cp-install.sh --rake='vcloud:resync'
```

Usage:

```
/onapp/onapp-cp-install/onapp-cp-install.sh [-c CONFIG_FILE] [--mariadb | --community | --percona | --percona-cluster] [-m MYSQL_HOST] [--mysql-port=MYSQL_PORT] [--mysql-sock[=MYSQL_SOCK] [-p MYSQL_PASSWD] [-d MYSQL_DB] [-u MYSQL_USER] [-U ADMIN_LOGIN] [-P ADMIN_PASSWD] [-F ADMIN_FIRSTNAME] [-L ADMIN_LASTNAME] [-E ADMIN_EMAIL] [-v ONAPP_VERSION] [-i SNMP_TRAP_IPS] [--redis-host=REDIS_HOST] [--redis-bind[=REDIS_BIND] [--redis-passwd[=REDIS_PASSWD] [--redis-port=REDIS_PORT] [--redis-sock[=REDIS_SOCK] [--rbthost RBT_HOST] [--vcdlogin VCD_LOGIN] [--vcdpasswd VCD_PASSWD] [--vcdvhost VCD_VHOST] [--rbtlogin RBT_LOGIN] [--rbtpasswd RBT_PASSWD] [-a] [-y] [-D] [-t] [--noservices] [--ha-install] [--rake=RAKE_TASKS] [-h]
```

Where:	
Database server options:	**Default database SQL server is MySQL Server. Please use one of the following option to install LOCALLY.**
--mariadb	**MariaDB Server**
--community	**MySQL Community Server**
--percona	**Percona Server**
--percona-cluster	**Percona Cluster**
MYSQL_*	**Options are useful if MySQL is already installed and configured.**
-m MYSQL_HOST	**MySQL host. Default is 'localhost'**
--mysql-port=MYSQL_PORT	**TCP port where MySQL Server serves connections.** **Default values is 3306 for the local installation.**
--mysql-sock [=MYSQL_SOCK]	**Unix socket on which MySQL Server serves connections.** **Default values is /var/lib/mysql/mysql.sock. Used if local server only. The socket is unset if the option's argument isn't specified.**
-p MYSQL_PASSWD	**MySQL password. Random is generated if is not set or specified.**
-d MYSQL_DB	**OnApp MySQL database name. Default is 'onapp'**
-u MYSQL_USER	**MySQL user. Default is 'root'**
REDIS_*	**Options are useful if Redis Server is already installed and configured.**

--redis-host=REDIS_HOST	IP address/FQDN where Redis Server runs. It is used by Control Panel to connect to Redis Server. The Redis Server will be installed and configured on the current box if localhost/127.0.0.1 or box's public IP address (listed in SNMP_TRAP_IPS) is specified. Default value is 127.0.0.1. If local Redis, it will serve as well on the unix socket 'PORT' (if --redis-sock without argument isn't specified)
--redis-bind [=REDIS_BIND]	The IP address for Redis Server to serve connections (to listen). The option isn't mandatory.
--redis-port=REDIS_PORT	Redis Server listen port. Defaults are: 0 - if local server 6379 - if remote server
--redis-passwd [=REDIS_PASSWD]	Redis Server password to authentificate. Random password is generated if the option's argument isn't specified. By default no password is used for local Redis.
--redis-sock=REDIS_PATH :	Path to the Redis Server's socket. Used if local server only. Default is /tmp/redis.sock The socket is unset if the option's argument isn't specified.
ADMIN_*	Options are used to configure OnApp Control Panel administrator data. Please note, that these options are for NEW INSTALL only and not for upgrade
-P ADMIN_PASSWD	CP administrator password
-F ADMIN_FIRSTNAME	CP administrator first name
-L ADMIN_LASTNAME	CP administrator last name
-E ADMIN_EMAIL	CP administrator e-mail
--rbthost RBT_HOST	IP address/FQDN where RabbitMQ Server runs. The RabbitMQ will be installed and configured on the current box if localhost/127.0.0.1 or box's public IP address (enlisted in SNMP_TRAP_IPS). Default values are 127.0.0.1.
VCD_*	Options are usefull if vCloud/RabbitMQ are already installed and configured.
--vcdlogin VCD_LOGIN	RabbitMQ/vCloud user. Default value is 'rbtvcd'.
--vcdpasswd VCD_PASSWD	RabbitMQ/vCloud user password. The random password is generated if isn't specified.
--vcdvhost VCD_VHOST	RabbitMQ/vCloud vhost. Default value is '/'
RBT_*	Options are used to configure RabbitMQ manager account. If local RabbitMQ server.
--rbtlogin RBT_LOGIN	RabbitMQ manager login. The default value is 'rbtmgr'.
--rbtpasswd RBT_PASSWD	RabbitMQ manager password. The random password is generated if isn't specified.
--ha-install	Proceed with Control Panel and Hight Availability components installation
--rake RAKE_TASKS	List of OnApp Control Panel rake tasks (separated with space) to run at the very end of install or upgrade
-v ONAPP_VERSION	Install custom OnApp CP version

-i SNMP_TRAP_IPS	**IP addresses separated with coma for snmp to trap**
-y	**update OS packages (except of OnApp provided) on the box with 'yum update'.**
-a	**Do not be interactive. Process with automatic installation. Please note, this will continue OnApp Control Panel install/upgrade even if there is transaction currently running.**
-t	**Add to the database and download Base Templates. For new installs only.**
--noservices	**Do not start OnApp services: monit, onapp and httpd** **Please note, crond and all OnApp's cron tasks remain running. They could be disabled by stopping crond service manually for your own risk.**
-D	**do not make database dump, and make sure it is disabled in the cron and not running at the moment**
-c CONFIG_FILE	**Custom installer configuration file. Otherwise, preinstalled one is used.**
-h	**print this info**

You may wish to reboot your Control Panel server to take advantage of a new kernel if it is installed. It is not required immediately as a part of the upgrade process though.

> ⚠ Customers running vCloud integration pay attention that upgrade process can take some time, as vCloud and OnApp synchronization is running along with OnApp upgrade.

Perform the following steps if you plan to deploy Accelerator. Otherwise skip.

8. See the installer options below for details.
9. Specify user name and password for rabbitmq-server:

```
rabbitmqctl add_user username 'userpass'
```

10. Set permissions for this user:

```
rabbitmqctl set_permissions -p '/' username ".*" ".*" ".*"
```

11. Restart OnApp service:

```
service onapp restart
```

> ⚠ Compute Resources and Control Panel must use the same rabbitmq-server.

Upgrade CloudBoot Packages

> ⓘ Create a backup of the /tftpboot directory in case storage packages rollback will be needed.

To upgrade the OnApp Storage packages:

1. Upgrade the repo:

```
CP_host#> rpm -Uvh http://rpm.repo.onapp.com/repo/onapp-repo-4.3.noarch.rpm
```

2. Upgrade the packages:

```
CP_host#> yum update onapp-store-install
```

3. Run the script:

```
CP_host#> /onapp/onapp-store-install/onapp-store-install.sh
```

> ⚠ Be aware that the disk-less nodes password is the root password for the CloudBoot compute resources. By default it is blank.

When run in the interactive mode, enter the required information.

Upgrade CloudBoot Backup Servers

> ⓘ Make sure to update CloudBoot packages before proceeding to the upgrade of CloudBoot backup servers.

CloudBoot backup servers are CloudBooted KVM compute resources that can be be used as backup servers. The CloudBoot backup server upgrade procedure is almost the same as the CloudBoot compute resource upgrade. Follow the instructions provided in this section to upgrade CloudBoot backup servers in your cloud.

Once you have upgraded the CloudBoot dependencies, you have to reboot your Cloud Boot compute resource to update the Cloud Boot RPM. You do not need to perform any backup server upgrade operations using console.

To do so:

1. Go to your Control Panel **Settings** menu.
2. Click the **Compute resources** icon.
3. Click the label of the CloudBoot compute resource the backup server is based on.
4. On the compute resource details screen, click the **Actions** button, then click **Reboot Compute resource.**
5. A new screen will open asking for confirmation before reboot:
    - **Are you sure you want to reboot this compute resource?** Confirm that you want the compute resource to reboot.
6. When you're certain you want to proceed with the reboot, click the **Reboot** button.
7. Repeat these steps for all CloudBoot backup servers in your cloud.
8. Once all are rebooted, proceed to CloudBoot compute resources upgrade.

Upgrade CloudBoot Compute Resources

Depending on the infrastructure, scale and needs of your cloud we suggest the following methods of upgrading CloudBoot compute resources:

Simple Reboot	This method is the simplest method technically. It also ensures all tools are updated. However, it will result in some limited downtime (its duration depends on how many virtual servers are running on each compute resource).
Migrate and reboot	This method involves migrating all virtual servers off each CloudBoot compute resource in turn. The compute resource can then be safely rebooted, picking up the upgraded Integrated Storage and CloudBoot packages. Virtual servers that do not support hot migrate will have to be stopped.
Live Upgrade	This method will upgrade Integrated Storage components but will not upgrade CloudBoot image.

In case you have applied any custom configuration to your CloudBoot servers, it is recommended to recheck that this customization does not break new cloud boot image version. For this, reboot a compute resource and run Storage Health Check and Network Health Check. Make sure that Vdisks hosted on a compute resource are redundant and healthy before rebooting a CloudBoot compute resource.

Simple Reboot

Follow the below procedure to upgrade the CloudBoot compute resources with reboot:

1. Upgrade CloudBoot Packages.
2. When the CloudBoot packages upgrade is complete, stop all virtual servers which reside on the CloudBoot compute resources.
3. Reboot all CloudBoot compute resources.

Once the compute resources are booted, the upgrade is complete. Before starting all Virtual Servers please ensure that the diagnostics page does not report any issue. In case of any issue, please press repair button to resolve it, then continue with starting Virtual Servers.

> ⓘ Note that virtual servers cannot be stopped simultaneously, but must be stopped in sequence. This can result in considerable downtime if there are a large number of virtual servers.

Migrate and reboot

Use this procedure if you prefer migrating all virtual servers to another compute resource and conducting overall upgrade of your CloudBoot and Integrated Storage. Virtual servers that do not support hot migrate will have to be stopped.

Once you have upgraded the CloudBoot packages, you have to reboot your CloudBoot compute resources to update them.

To do so:

1. Run the following command from the Control Panel server terminal to display the list of compute resources with their IP addresses. Make a note of the list of IPs:

   ```
   CP_host#> liveUpdate listHVs
   ```

   This command will also show whether compute resources are eligible for live upgrade.

   > ⓘ If the command liveUpdate is not available then it may be located in the sbin directory instead (cd /usr/local/sbin).

2. Run the following command for every compute resource:

   ```
   CP_host#> liveUpdate updateToolstack <HV IP Addr>
   ```

   Once all the toolstacks are updated run the following command for every compute resource:

   ```
   CP_host#> liveUpdate refreshControllers <HV IP Addr>
   ```

   > ⓘ Wait several minutes for all degraded disks to come to synchronized state. The synchronization will take approximately three minutes for each compute resource.
   >
   > After each controller restart, check for any issues on the backup server (or on one Compute resource from each zone):
   >
   > 1. Log on via SSH to the backup server (or Compute resource).
   > 2. Run getdegradednodes from the SSH console.
   > 3. Run getdegradedvdisks from the SSH console.

3. Migrate all the virtual servers from the CloudBoot compute resource to another compute resource. Follow the instructions described in the Migrate Virtual Server section of the Admin guide to migrate virtual servers.
4. After that, go to your Control Panel **Settings** menu.
5. Click the **Compute Resources** icon.
6. Click the label of the CloudBoot compute resource you have migrated all VSs from.
7. On the compute resource details screen, click the **Actions** button, then click **Reboot Compute resource.**

   > ⓘ Rebooting a compute resource assigned to a data store with a single replica (single-replica compute resource) or degraded virtual disks may result in data loss.

8. A new screen will open asking for confirmation (via two check boxes) before reboot:

   - **Stop all virtual servers that cannot be migrated to another compute resource?** Check this box if you want VSs that cannot be migrated to be powered off. When a compute resource is scheduled for a reboot, OnApp will first attempt to hot migrate all VSs it hosts. If hot migration is not possible for a VS, OnApp will attempt to cold migrate that VS. With this box checked, if cold migration fails, the VS will be stopped so the reboot may proceed. If you don't check this box, OnApp will attempt to hot and then cold migrate all VSs hosted by the compute resource being rebooted – but will stop the migration process if any VS cannot be migrated.
   - **Are you sure you want to reboot this compute resource?** A simple confirmation to confirm that you want the compute resource to reboot.

   > Before the reboot, please ensure that all vdisks are fully synced and redundant. If some of them are not fully synced, the virtual server, that is owner of a degraded (or non-redundant) vdisk, can loose access to the vdisk. It can be manifested as IO errors during writes or reads to/from the vdisk inside the virtual server.

9. When you're certain you want to proceed with the reboot, click the **Reboot** button.
10. Once the compute resource is booted, repair the disk that were degraded during the reboot.

    a. Make sure no disks are out of sync. To do so, check the Diagnostics page in CP at **Dashbord** > **Integrated Storage** > **Compute zonelabel** > **Diagnostics**. Alternatively, log into a compute resource and run the command below:

    ```
    HV_host#> getdegradedvdisks
    ```

    b. Repair all the degraded disks before proceeding to the upgrade process. To do so, log in to your CP and go to **Integrated Storage** > **Compute zonelabel** > **Diagnostics** page. Alternatively, run one of the following commands:

    ```
    HV_host#> onappstore repair uuid=
    ```

    ```
    HV_host#> parallelrepairvdisks
    ```

11. Repeat these steps for all CloudBoot compute resources in your cloud.

Live Upgrade

> Live Upgrade is only applicable if your cloud is running latest 4.3 CloudBoot RPM.

- Live Upgrade with passthrough is currently unsupported. Passthrough to storage means that network interface will be added to the Storage Controller Server without the bond and the Storage Controller Server will have the complete control over this interface.
- Power off all Windows virtual machines and virtual backup servers before starting the live upgrade.
- If your current Storage package is 4.0, Windows virtual servers can remain running.
- During the CloudBoot compute resource live upgrade, only the control stack for managing integrated storage is upgraded. Other changes come into effect after the compute resource is next rebooted. Due to this, hot migration may fail between compute resource which is already rebooted and the one that hasn't.
- Do not make any changes to the cloud during the upgrade process!
- Any offline Cloudboot compute resources should be removed from the CP server before running live upgrade as the scripts expect to be able to speak to all compute resources during these steps.
- Please, consult OnApp IS Upgrade Paths to learn the minimum Integrated Storage version required for the current update to be performed in LiveUpgrade mode.

Use this procedure to upgrade without rebooting your servers:

1. Make sure no disks are out of sync. To do so, check the Diagnostics page in CP at **Dashbord** > **Integrated Storage** > **Compute zonelabel** > **Diagnostics**. Alternatively, log into a compute resource and run the command below:

   ```
   HV_host#> getdegradedvdisks
   ```

2. Repair all the degraded disks before proceeding to the upgrade process. To do so, log in to your CP and go to **Integrated Storage** > **Compute zonelabel** > **Diagnostics** page. Alternatively, run one of the following commands:

```
HV_host#> onappstore repair uuid=
```

```
HV_host#> parallelrepairvdisks
```

3. Run the following command from the CP server to stop the OnApp service:

   ```
   CP_host#> service onapp stop
   ```

4. Stop the Apache server:

   ```
   CP_host#> service httpd stop
   ```

5. Make sure to update CloudBoot packages before proceeding to the following steps.
6. Run the following command from the Control Panel server terminal to display the list of compute resources with their IP addresses. Make a note of the list of IPs:

   ```
   CP_host#> liveUpdate listHVs
   ```

   This command will also show whether compute resources are eligible for live upgrade.

   > If the command liveUpdate is not available then it may be located in the sbin directory instead (cd /usr/local/sbin).

7. Run the following command for every compute resource:

   ```
   CP_host#> liveUpdate updateToolstack <HV IP Addr>
   ```

   Once all the toolstacks are updated run the following command for every compute resource:

   ```
   CP_host#> liveUpdate refreshControllers <HV IP Addr>
   ```

   > Wait several minutes for all degraded disks to come to synchronized state. The synchronization will take approximately three minutes for each compute resource.
   >
   > After each controller restart, check for any issues on the backup server (or on one Compute resource from each zone):
   >
   > 1. Log on via SSH to the backup server (or Compute resource).
   > 2. Run getdegradednodes from the SSH console.
   > 3. Run getdegradedvdisks from the SSH console.

8. Restarts the storage controllers. This command can be **performed later at a more suitable time**.
   Run the following command for each compute resource in turn:

   ```
      CP_host#> liveUpdate restartControllers <HV IP Addr>
   ```

   > Please make sure you restart all controllers and don't leave your cloud in a partially updated state for too long. Note that when operating in LiveUpdated mode (e.g. with the tool stacks updated but before you have performed the controller restart) you cannot use disk hot plug.

   > After each controller restart check for any issues on the backup server or one Hypervisor from each zone:
   > 1. Log on via SSH to the backup server (or Hypervisor).
   > 2. Run getdegradednodes from the SSH console.
   > 3. Run getdegradedvdisks from the SSH console.
   >
   > If there are any issues seen please rectify them before continuing with the next controller restart.

9. Make sure that the package versions are upgraded by running the following command on each compute resource:

```
HV_host#> cat /onappstore/package-version.txt | grep Source
```

10. Start the Apache server:

    ```
    CP_host#> service httpd start
    ```

11. Start the OnApp service:

    ```
    CP_host#> service onapp start
    ```

Configuration for Accelerator

Perform the following steps for your CloudBoot compute resources if you plan to deploy Accelerator. These steps are to be performed on each of the compute resources.

1. Copy file:

    ```
    cp /home/mq/onapp/messaging/credentials{_example,}.yml
    ```

2. Open `vi /home/mq/onapp/messaging/credentials.yml` and check the following details:

    ```
    ---
    host: 10.0.50.4  # RABBITMQ SERVER IP/FQDN
    port: 5672          # RABBITMQ CONNECTION PORT(default: 5672)
    vhost: '/'
    user: accelerator-example # RABBITMQ USER NAME
    password: 'e{y31?s8l' # RABBITMQ ACCESS PASSWORD
    queue: 'hv-10.0.50.102' # hv-[IP Address of Compute Resource]
    exchange:
      name: 'acceleration'
      type: 'direct'
      durable: True
    ```

3. Change owner:

    ```
    chown -R mq:mq /home/mq
    ```

4. Run the following:

    ```
    service onapp-messaging start
    ```

Note that steps 1-4 of the above instruction should be done after every reboot of CloudBoot compute resource. You can run the following commands (using your parameters) to the custom config instead:

```
cp /home/mq/onapp/messaging/credentials{_example,}.yml
echo "---
host: 10.0.50.4# RABBITMQ SERVER IP/FQDN
port: 5672# RABBITMQ CONNECTION PORT(default: 5672)
vhost: '/'
user: accelerator-example # RABBITMQ USER NAME
password: 'e{y31?s8l' #RABBITMQ ACCESS PASSWORD
queue: 'hv-10.0.50.102' # hv-[IP Address of Compute Resource]
exchange:
name: 'acceleration'
type: 'direct'
durable: True" > /home/mq/onapp/messaging/credentials.yml
chown -R mq:mq /home/mq
service onapp-messaging restart
```

Also you can monitor RabbitMQ server using a command-line tool for it.

Local Read Policy

Enabling Local Read on a compute zone ensures that the locally stored copy of the data will always be used for reads. This significantly reduces read latency and improves overall storage performance by reducing load on the SAN network. However, in order to use this policy every compute resource must have sufficient physical drives to be able to store the number of stripes specified in the data store. E.g. in a 2R4S data store there must be at least 4 physical disks on the compute resource to use local read.

Changes to Local Read Policy Enforcement

Originally, when this policy was introduced OnApp did not enforce the requirement for the minimum number of drives. Consequently, some users who set the policy having insufficient drives may see the following error message:

```
Fatal: OnApp::Actions::Fatal Storage API Call failed: {"result"=>"FAILURE", "error"=>"Local reads have been enabled on the zone - members required per host: 4, required hosts: 2, available hosts: 0"}
```

The solution is to either add additional drives to that compute resource and then add them to the data store or to disable read local.

Getting support for your upgrade

You can use the information in this document to perform your own upgrade to the 4.3 version of the OnApp Cloud. However, if you have a full OnApp Cloud license, you are entitled to free upgrade support from the OnApp Support team.

If you would prefer to have the Support team perform the upgrade for you, just raise a ticket in the normal way. Please be aware, however, that there may be a queue! For help with your upgrade, visit the OnApp community forum: http://forum.onapp.com.