Chapter 18

Backup & Restore, Upgrades, DNS / NTP Server Migration

This document provides details on how to perform backup and restore operations and upgrading CloudVision Portal (CVP).

- Backup and Restore on page 324
- Upgrading CloudVision Portal (CVP) on page 326
- DNS / NTP Server Migration on page 334
18.1 Backup and Restore

CloudVision Portal (CVP) enables you to backup and restore the complete CVP provisioning dataset, including containers, devices, configlets, images, and configlet/image assignments. You can use commands to backup and restore CVP data.

The CVP commands provide all of the functionality required to complete backup and restore operations.

**Important!**

It is a good practice to regularly create and export backups to ensure that you have an adequate supply of backup files available to you that you can use to restore CVP data.

**Note**

There is no backup or restore of the Telemetry analytics dataset.

The 2018.2 release does not support restoring backups taken from pre 2018.2 releases. If you would like to restore a backup from a previous release, install 2018.1.2+, restore the backup, and then fast upgrade to 2018.2. After you have successfully upgraded to 2018.2, take another backup so that you can directly restore that into a 2018.2 release in the future.

For more information, see:

- Requirements for Multi-node Installations
- Using CVPI Commands to Backup and Restore Data

18.1.1 Requirements for Multi-node Installations

The basic requirements for backup and restore operations are the same for single-node installations and multi-node installations.

**Note**

For multi-node CVP installations, you do not have to backup and restore data on all of the nodes in the cluster. The backup and restore process needs to be completed on just the Primary node of the cluster.

18.1.2 Using CVPI Commands to Backup and Restore Data

You can use CVPI commands to backup and restore CVP data.

**Note**

Arista provides a simple script at /cvpi/tools/backup.py which is scheduled by default to run daily to backup CVP data, and retain the last 5 backups in /data/cvpbackup/. Backing up and restoring data saves information about the CVP instance to a tar file, and then restores the information from the tar file to a new CVP instance.

18.1.2.1 Using CVPI Commands to Backup and Restore CVP Provisioning Data

The following commands can be used to backup and then restore the containers, devices, configlets, images, and configlet/image assignments that you have defined in CVP.

The default directory for backup data files is /data/cvpbackup. The default directory for backup/restore log files is /cvpi/logs/cvpbackup. The default directory for temporary files during backup/restore is /data/tmp/cvpbackup.
Note

When restoring devices, use a username and password that can access the devices you are registering.

For cvp component backup use the following command:

```
[cvp@cvp108 bin]$ cvpi backup cvp
```

To restore cvp component, use the following command:

```
[cvp@cvp108 bin]$ cvpi restore cvp <cvp tgz> [<eosimages tgz>]
```

cvp.tgz: contains the DB data for the CVP application

eosimages.tgz: contains the HDFS contents for the CVP application (example: EOS swi and swix)
18.2 Upgrading CloudVision Portal (CVP)

Similar to Arista EOS, CVP is packaged and released in trains. Refer to the latest release notes and upgrade procedures while upgrading CVP.

The 2018.2 release has the following new requirements for devices under management. Devices must:

- be running EOS 4.17+
- have TerminAttr 1.4.1+ installed;
- have the TerminAttr agent enabled and successfully streaming telemetry to CVP.

The following steps can be taken at any point on an existing cluster as part of preparing for an upgrade to 2018.2.

**Step 1** Upgrade existing CVP clusters to the latest 2018.1 release (currently 2018.1.4)

**Step 2** Upgrade all EOS devices under management to a 4.17+ release train

**Step 3** Disable DCA (Beta feature that is disabled by default) if previously enabled

**Step 4** For devices running EOS releases prior to 4.20, ensure that the eAPI unix domain socket is enabled with the following configuration:

```
management api http-commands
protocol unix-socket
```

**Step 5** Install TerminAttr 1.4.1+ on all EOS devices under management

**Step 6** Enable state streaming from all EOS devices under management by applying the SYS_StreamingTelemetry configlet and pushing the required configuration to all devices

**Step 7** Ensure that all devices are successfully streaming to the CVP cluster

**Step 8** Ensure that all devices are in image and config compliance

**Step 9** Take regular backups / take a final backup prior to upgrade

If you have a system running TerminAttr before 1.4.1, after the upgrade to 2018.2, the following features will not work correctly on those devices:

- Viewing the device running-config
- Device running-config snapshot and rollback
- Device image flavor (ie PDP, 2GB, etc) snapshot and rollback

Upgrading TerminAttr to a supported version resolves this.

**Note**

You cannot rollback a device to a time before it was running TerminAttr 1.4.1+.

Upgrading a system from a pre 2018.2 release to a post 2018.2 release will require a cluster reboot to complete the upgrade of the base OS to Centos 7.5.

The upgrade from the 2018.1 to 2018.2 trains includes data migrations that can take several hours on larger scale systems.

- **Upgrades on page 327**
- **CVP Node RMA on page 328**
- **CVP / EOS Dependencies on page 333**
18.2.1 Upgrades

Upgrades do not require that the VMs be redeployed, and do not result in the loss of logs. The CVP cluster must be functional and running to successfully complete an upgrade. As a precaution against the loss of CVP data, it is recommended that you back-up the CVP data before performing an upgrade.

Upgrade to the CloudVision Portal 2018.2 train is supported from existing systems running 2018.1.2 or above. Clusters running earlier release trains should first be upgraded to the latest 2018.1 release (currently 2018.1.4) before upgrading to the 2018.2 train.

Verifying the health of CVP before performing upgrades

Upgrades should only be performed on healthy and fully functional CVP systems. Before performing the upgrade, make sure that you verify that the CVP system is healthy.

Complete the following steps to verify the health of CVP.

**Step 1** Enter into the linux shell of the primary node as `cvp user`.

**Step 2** Execute the `cvpi status all` command on your CVP:

This shows the status of all CVP components.

**Step 3** Confirm that all CVP components are running.

**Step 4** Log into the CVP system to check functionality.

Once you have verified the health of your CVP installation, you can begin the upgrade process.

- **Upgrading from version 2018.1.2 (or later)**

18.2.1.1 Upgrading from version 2018.1.2 (or later)

Use this procedure to complete the fast upgrade of CVP to the current version of CVP.

**Pre-requisites:**

Before you begin the upgrade procedure, make sure that you have:

- Verified the health of your CVP installation (see Verifying the health of CVP before performing upgrades).
- Verified that you are running version 2018.1.2 or later.

Complete the following steps to perform the upgrade.

**Step 1** SSH as root into the primary node.

**Step 2** Run these commands:

a. `cd /tmp/`

b. `mkdir upgrade`

c. `cd upgrade`

d. `rm * -f` (to remove data from old upgrades if already present)

e. `scp/wget cvp-upgrade-<version>.tgz` to this directory.

**Step 3** Run the `su cvpadmin` command to trigger the shell.

**Step 4** Select the upgrade option from the shell.
Note
On a multi-node cluster, upgrade can be performed only on the primary node.
Upgrading to the current version may take up to 30 minutes.

Related topics:
- CVP Node RMA
- CVP / EOS Dependencies on page 333
- Upgrades on page 327

18.2.2 CVP Node RMA

Use this procedure to replace any node of a multi-node cluster. Replacing nodes of multi-node cluster
involves removing the node you want to replace, waiting for the remaining cluster nodes to recover,
powering on the replacement node, and applying the cluster configuration to the new node.

When you replace cluster nodes, you must replace only one node at a time. In case, you plan to
replace more than one node of a cluster, you must complete the entire procedure for each node to be
replaced.

Note
It is recommended that you save the cvp cluster configuration to a temporary file, or write down the
configuration on a worksheet. The configuration can be found in /cvpi/cvp-config.yaml.

Step 1 Power off the node you want to replace (primary, secondary, or tertiary).
Step 2 Remove the node to be replaced.
Step 3 Allow all components of the remaining nodes to recover.
Step 4 Use the cvp status all command to ensure that remaining nodes are healthy.

Example
[cvp@cvp73 root]$ cvpi status all

Current Running Command: None
Executing command... This may take a few seconds...
primary 78/78 components running
secondary 89/89 components running
tertiary NODE DOWN

Step 5 Power on the replacement node.
Step 6 Log in as “cvpadmin”.
Step 7 Enter the cvp cluster configuration.
Example

CentOS Linux 7 (Core)
Kernel 3.10.0-957.1.3.el7.x86_64 on an x86_64

localhost login: cvpadmin
Last login: Fri Mar 15 12:24:45 on ttyS0
Changing password for user root.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
Enter a command
> r
Please enter minimum configuration to connect to the other peers
*Ethernet interface for the cluster network: eth0
*IP address of eth0: 172.31.0.216
*Netmask of eth0: 255.255.0.0
*Default route: 172.31.0.1
*IP address of one of the two active cluster nodes: 172.31.0.161
Root password of 172.31.0.161:

Step 8  Wait for the RMA process to complete. No action is required.
Example

Root password of 172.31.0.161:
External interfaces, ['eth1'], are discovered under /etc/sysconfig/network-scripts
These interfaces are not managed by CVP.
Please ensure that the configurations for these interfaces are correct.
Otherwise, actions from the CVP shell may fail.

Running : /bin/sudo /sbin/service network restart

[ 334.001886] vmxnet3 0000:0b:00.0 eth0: intr type 3, mode 0, 9 vectors allocated
[ 334.004577] vmxnet3 0000:0b:00.0 eth0: NIC Link is Up 10000 Mbps
[ 334.006315] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready
[ 334.267535] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
[ 348.252323] vmxnet3 0000:13:00.0 eth1: intr type 3, mode 0, 9 vectors allocated
[ 348.254925] vmxnet3 0000:13:00.0 eth1: NIC Link is Up 10000 Mbps
[ 348.256504] IPv6: ADDRCONF(NETDEV_UP): eth1: link is not ready

Fetching version information
Run cmd: sudo -u cvp -- ssh 172.31.0.156 cat /cvpi/property/version.txt 0.18
Fetching version information
Run cmd: sudo -u cvp -- ssh 172.31.0.216 cat /cvpi/property/version.txt 10.19
Fetching version information
Run cmd: sudo -u cvp -- ssh 172.31.0.161 cat /cvpi/property/version.txt 0.16

Running : cvpConfig.py tool...

[ 392.941983] vmxnet3 0000:0b:00.0 eth0: intr type 3, mode 0, 9 vectors allocated
[ 392.944739] vmxnet3 0000:0b:00.0 eth0: NIC Link is Up 10000 Mbps
[ 392.946460] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready
[ 392.169460] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
[ 407.229180] vmxnet3 0000:13:00.0 eth1: intr type 3, mode 0, 9 vectors allocated
[ 407.232306] vmxnet3 0000:13:00.0 eth1: NIC Link is Up 10000 Mbps
[ 407.233940] IPv6: ADDRCONF(NETDEV_UP): eth1: link is not ready
[ 408.447642] Netfilter Core Team
[ 408.935626] ip_tables: (C) 2000-2006 Netfilter Core Team
[ 408.956578] ip6_tables: (C) 2000-2006 Netfilter Core Team
[ 408.982927] Netfilter Core Team

Stopping: ntpd
Running : /bin/sudo /sbin/service ntpd stop
Running : /bin/sudo /bin/systemctl is-active ntpd
Starting: ntpd
Running : /bin/sudo /bin/systemctl start ntpd.service
Waiting for all components to start. This may take few minutes.

Run cmd: su - cvp -c '/cvpi/bin/cvpi -v=3 status zookeeper' 0.45
Run cmd: su - cvp -c '/cvpi/bin/cvpi -v=3 status zookeeper' 0.33
Checking if third party applications exist
Run cmd: su - cvp -c '/cvpi/zookeeper/bin/zkCli.sh ls /apps | tail -1' 0.72

Running : cvpConfig.py tool...
Stopping: cvpi-check
Running : /bin/sudo /sbin/service cvpi-check stop
Running : /bin/sudo /bin/systemctl is-active cvpi-check
Starting: cvpi-check
Running : /bin/sudo /bin/systemctl start cvpi-check.service

Step 9  Continue waiting for the RMA process to complete. No action is required.
Example

[Fri Mar 15 20:26:28 UTC 2019] :
Executing command. This may take a few seconds...

(E) => Enabled
(D) => Disabled
(?) => Zookeeper Down

Action Output
-------------
<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>ACTION</th>
<th>NODE</th>
<th>STATUS</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>hadoop</td>
<td>cluster</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>hbase</td>
<td>cluster</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
</tbody>
</table>

Executing command. This may take a few seconds...

(E) => Enabled
(D) => Disabled
(?) => Zookeeper Down

Action Output
-------------
<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>ACTION</th>
<th>NODE</th>
<th>STATUS</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>aerisdiskmonitor</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>aerisdiskmonitor</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>aerisdiskmonitor</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>apiserver</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>apiserver</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>apiserver</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>cvp-backend</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>cvp-backend</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>cvp-backend</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>cvp-frontend</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>cvp-frontend</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>cvp-frontend</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>geiger</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>geiger</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>geiger</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>hadoop</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>hadoop</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>hadoop</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>hbase</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>hbase</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>hbase</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>kafka</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>kafka</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>kafka</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>zookeeper</td>
<td>config</td>
<td>primary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>zookeeper</td>
<td>config</td>
<td>secondary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
<tr>
<td>zookeeper</td>
<td>config</td>
<td>tertiary</td>
<td>(E)</td>
<td>DONE</td>
</tr>
</tbody>
</table>

Executing command. This may take a few seconds...

secondary 89/89 components running
primary 78/78 components running

Executing command. This may take a few seconds...

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>ACTION</th>
<th>NODE</th>
<th>STATUS</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Including: /cvpi/tls/certs/cvp.crt
Including: /cvpi/tls/certs/cvp.key
Including: /etc/cvpi/cvpi.key
Including: /cvpi/tls/certs/kube-cert.pem
Including: /data/journalnode/mycluster/current/VERSION
Including: /data/journalnode/mycluster/current/last-writer-epoch
Including: /data/journalnode/mycluster/current/last-promised-epoch
Including: /data/journalnode/mycluster/current/paxos
Including: /cvpi/tls/certs/ca.crt
Including: /cvpi/tls/certs/ca.key
Including: /cvpi/tls/certs/server.crt
Including: /cvpi/tls/certs/server.key
mkdir -p /cvpi/tls/certs
mkdir -p /data/journalnode/mycluster/current
mkdir -p /cvpi/tls/certs
mkdir -p /etc/cvpi
mkdir -p /cvpi/tls/certs
mkdir -p /cvpi/tls/certs
mkdir -p /cvpi/tls/certs
mkdir -p /data/journalnode/mycluster/current
mkdir -p /data/journalnode/mycluster/current
Copying: /etc/cvpi/cvpi.key from secondary
rsync -rtvp 172.31.0.161:/etc/cvpi/cvpi.key /etc/cvpi
Copying: /cvpi/tls/certs/cvp.crt from secondary
rsync -rtvp 172.31.0.161:/cvpi/tls/certs/cvp.crt /cvpi/tls/certs
Copying: /cvpi/tls/certs/server.key from secondary
rsync -rtvp 172.31.0.161:/cvpi/tls/certs/server.key /cvpi/tls/certs
Copying: /cvpi/tls/certs/ca.crt from secondary
rsync -rtvp 172.31.0.161:/cvpi/tls/certs/ca.crt /cvpi/tls/certs
Copying: /cvpi/tls/certs/cvp.key from secondary
rsync -rtvp 172.31.0.161:/cvpi/tls/certs/cvp.key /cvpi/tls/certs
Copying: /cvpi/tls/certs/ca.key from secondary
rsync -rtvp 172.31.0.161:/cvpi/tls/certs/ca.key /cvpi/tls/certs
Copying: /data/journalnode/mycluster/current/last-writer-epoch from secondary
rsync -rtvp 172.31.0.161:/data/journalnode/mycluster/current/last-writer-epoch /data/journalnode/mycluster/current
Copying: /cvpi/tls/certs/kube-cert.pem from secondary
rsync -rtvp 172.31.0.161:/cvpi/tls/certs/kube-cert.pem /cvpi/tls/certs
Copying: /data/journalnode/mycluster/current/server.crt from secondary
rsync -rtvp 172.31.0.161:/data/journalnode/mycluster/current/server.crt /data/journalnode/mycluster/current
Copying: /data/journalnode/mycluster/current/VERSION from secondary
rsync -rtvp 172.31.0.161:/data/journalnode/mycluster/current/VERSION /data/journalnode/mycluster/current
Copying: /data/journalnode/mycluster/current/paxos from secondary
rsync -rtvp 172.31.0.161:/data/journalnode/mycluster/current/paxos /data/journalnode/mycluster/current
Copying: /data/journalnode/mycluster/current/last-promised-epoch from secondary
rsync -rtvp 172.31.0.161:/data/journalnode/mycluster/current/last-promised-epoch /data/journalnode/mycluster/current
rsync -rtvp 172.31.0.161:/cvpi/tls/certs/kube-cert.pem /cvpi/tls/certs
Starting: cvpi-config
Running : /bin/sudo /bin/systemctl start cvpi-config.service
Starting: cvpi
Running : /bin/sudo /bin/systemctl start cvpi.service
Running : /bin/sudo /bin/systemctl start cvpi-watchdog.timer
Running : /bin/sudo /bin/systemctl enable docker
Running : /bin/sudo /bin/systemctl start docker
Running : /bin/sudo /bin/systemctl enable kube-cluster.path

Step 10 Enter "q" to quit the process after RMA process is complete! message is displayed.
Example

Waiting for all components to start. This may take few minutes.

[ 560.918749] FS-Cache: Loaded
[ 560.978183] FS-Cache: Netfs 'nfs' registered for caching

Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 48.20
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.73
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 7.77
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.55
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.23
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.64
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.59
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.07
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.57
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.40
Run cmd: su - cvp -c '/cvpi/bin/cvpi status all --cluster' 2.24
Waiting for all components to start. This may take few minutes.
Run cmd: su - cvp -c '/cvpi/bin/cvpi -v=3 status all' 9.68
RMA process is complete!


Example

[cvp@cvp87 ~]$ cvpi status all

Current Running Command: None
Executing command. This may take a few seconds...
primary 78/78 components running
secondary 89/89 components running
tertiary 45/45 components running

Related topics:
- CVP / EOS Dependencies
- Upgrades on page 327

18.2.3 CVP / EOS Dependencies

To ensure that CVP can provide a base level of management, all EOS devices must be running at least EOS versions 4.17.3F or later. To ensure device compatibility supported EOS version advice should be sought from the Arista account team.

CVP should not require any additional EOS upgrades to support the standard features and functions in later versions of the appliance. Newer features and enhancements to CVP may not be available for devices on older code versions.

Refer to the latest Release Notes for additional upgrade/downgrade guidance.

Related topics:
- Upgrades on page 327
- CVP Node RMA on page 328
18.3 DNS / NTP Server Migration

You can migrate your DNS / NTP server after you have completed your initial deployment of CloudVision. Migrating the DNS / NTP server is typically done if you want to or need to change the DNS / NTP server that CloudVision currently uses.

For example, if the current CloudVision DNS / NTP server was intentionally isolated during the initial CloudVision installation, you need to migrate the server to make it accessible by external resources.

18.3.0.1 Migrating the DNS and NTP Server

The process for migrating the DNS / NTP server after the completion of the initial CloudVision installation involves updating the DNS and NTP server entries on each cluster node and modifying the /cvpi/cvp-config.yaml file (on each node) to reflect the updates to the server entries.

Pre-requisites

Before you begin the migration process, make sure that:

- The IP addresses and hostnames (fqdn) of the nodes must not change.
- For each node, make sure that:
  - At least one DNS server entry is present in the /cvpi/cvp-config.yaml file.
  - The DNS server that corresponds to the DNS server entry in the /cvpi/cvp-config.yaml file can be accessed by the cluster throughout the migration process. (The reason for this is that any changes made to resolv.conf take effect immediately upon saving the file.)
  - The time difference between the old NTP server and new NTP server should be negligible.
  - The old NTP server and new NTP server should be in same time zone.

Complete these steps to migrate the DNS / NTP server.

Step 1  On each node, add the new server to /etc/resolv.conf, by adding a new nameserver line at the top of the file. For example, nameserver 172.22.22.40.

Step 2  On each node, remove the old server from /etc/resolv.conf, by removing the old nameserver line.

Step 3  On each node, do the following to update the NTP server:
  a. Run the ntpstat command to make note of the current NTP server.
  b. In /etc/ntp.conf, add the new NTP server entry and comment out the entry for the old NTP server.
  c. Run the service ntpd restart command.
  d. Run the ntpstat command to verify that the NTP server has been changed on all nodes.

Step 4  On each node, edit the /cvpi/cvp-config.yaml file to reflect the changes to the DNS and NTP server entries you made in the previous steps.

Related topics:

- Backup and Restore on page 324
- Upgrading CloudVision Portal (CVP) on page 326