Oracle Private Cloud Appliance

Patching Guide

F74800-09
April 2024

Oracle Private Cloud Appliance Patching Guide,

F74800-09

Contents

Preface

This publication is part of the customer documentation set for Oracle Private Cloud Appliance Release 3.0. Note that the documentation follows the release numbering scheme of the appliance software, not the hardware on which it is installed. All Oracle Private Cloud Appliance product documentation is available at https://docs.oracle.com/en/engineered-systems/private-cloud-appliance/index.html.

Oracle Private Cloud Appliance Release 3.x is a flexible general purpose Infrastructure as a Service solution, engineered for optimal performance and compatibility with Oracle Cloud Infrastructure. It allows customers to consume the core cloud services from the safety of their own network, behind their own firewall.

Audience

This documentation is intended for owners, administrators and operators of Oracle Private Cloud Appliance. It provides architectural and technical background information about the engineered system components and services, as well as instructions for installation, administration, monitoring and usage.

Oracle Private Cloud Appliance has two strictly separated operating areas, known as enclaves. The Compute Enclave offers a practically identical experience to Oracle Cloud Infrastructure: It allows users to build, configure and manage cloud workloads using compute instances and their associated cloud resources. The Service Enclave is where privileged administrators configure and manage the appliance infrastructure that provides the foundation for the cloud environment. The target audiences of these enclaves are distinct groups of users and administrators. Each enclave also provides its own separate interfaces.

It is assumed that readers have experience with system administration, network and storage configuration, and are familiar with virtualization technologies. Depending on the types of workloads deployed on the system, it is advisable to have a general understanding of container orchestration, and UNIX and Microsoft Windows operating systems.

Feedback

Provide feedback about this documentation at https://www.oracle.com/goto/docfeedback.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

Convention	Meaning
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, code in examples, text that appears on the screen, or text that you enter.
`$` prompt	The dollar sign (`$`) prompt indicates a command run as a non-root user.
`#` prompt	The pound sign (`#`) prompt indicates a command run as the `root` user.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

Patching Your Oracle Private Cloud Appliance

This document describes the patching process for your Oracle Private Cloud Appliance. Upgrading your appliance is a different process, refer to the Oracle Private Cloud Appliance Upgrade Guide for those directions.

Starting with release 3.0.1, Oracle Private Cloud Appliance supports patching updates for security fixes and software errata between major releases. To take advantage of this feature you must configure your environment to support channel updates.

Patches are delivered as RPM packages through a series of dedicated channels on the Unbreakable Linux Network (ULN). To gain access to these channels, you need a Customer Support Identifier (CSI) and a ULN subscription.

Oracle Private Cloud Appliance is not allowed to connect directly to Oracle ULN servers. You must use a ULN mirror on a system inside the data center. The patch channels are then synchronized on the ULN mirror, where the management nodes can access the RPMs. Compute nodes need access to a subset of the RPMs, which are copied to a designated location on the appliance internal shared storage and kept up-to-date.

Patching Strategy
We recommend to run the latest available software on your Oracle Private Cloud Appliance. It improves protection against vulnerabilities and allows you to take advantage of all new features, bug fixes, and functional improvements.
The latest Upgrader code automatically enforces prerequisite software versions. During the upgrade or patch preparations, the Upgrader service validates the currently installed appliance software version against the new target version. If the appliance is not running at least the minimum required version, the Upgrader exits the process and rolls back the environment to its previous state. You must first install the prerequisite version as indicated in the log.

Patching Order
Components must be patched in a prescribed order. In appliance software version 3.0.2-b892153 and later, the upgrade plan helps manage the order of patch operations. When patching to version 3.0.2-b1081557 or later, there is an extra requirement to patch the ZFS Storage Appliance firmware before all other components. For more information, see Checking Upgrade Plan Status and Progress.

2
Configure Your Environment for Patching

To be able to patch an Oracle Private Cloud Appliance with RPM packages released through the Unbreakable Linux Network (ULN), you must set up a ULN mirror in your data center and configure the appliance to use it as a source of updates. Both the Private Cloud Appliance and the ULN mirror must be registered with ULN under your Customer Support Identifier (CSI). One mirror can be used by multiple Private Cloud Appliance systems.

This chapter contains references to CSI administration, ULN registration, and ULN mirror setup, which are described in detail in the Oracle Linux documentation.

- Managing Software in Oracle Linux, a guide applicable to Oracle Linux 8 and Oracle Linux 9
- Unbreakable Linux Network User's Guide for Oracle Linux 6 and Oracle Linux 7

To set up a ULN mirror for Private Cloud Appliance patching, you can use Oracle Linux 8 or Oracle Linux 7. There are minor differences, which are covered in this chapter. However, as Oracle Linux 7 enters extended support in December 2024, we recommend that new ULN mirrors be set up on Oracle Linux 8.

Complete Your ULN Registration

To register your Private Cloud Appliance and ULN mirror system, proceed as follows:

1. Obtain a valid Customer Support Identifier (CSI).

   Your CSI was issued to you when you purchased the appliance. For more information, see Managing ULN Users.

2. Set up a system to use as ULN mirror. An installation of Oracle Linux 8 is preferred, but an existing Oracle Linux 7 machine can also be used. Package repository configuration is explained further in this chapter.

3. Register both your mirror server and Oracle Private Cloud Appliance with ULN. See Registering to Use ULN and How to Register a System With ULN.

Configure the Local ULN Mirror

The purpose of the ULN mirror is to replicate channels from the Unbreakable Linux Network (ULN) locally so that your Private Cloud Appliance can access the packages required for patching.

1. Set up the local ULN mirror system as a yum server and enable replication for the required ULN channels. Refer to the instructions that correspond to the installed operating system.

   - Oracle Linux 8: Setting up a Local ULN Mirror
   - Oracle Linux 7: Setting up a Local ULN Mirror

> **Note:**
>
> For consistency with Oracle Linux 7 mirrors, we recommend to continue using `uln-yum-mirror` on Oracle Linux 8, instead of the `dnf reposync` command.
>
> It is considered good practice to keep the operating system of the mirror server up-to-date.

2. Reserve approximately 60GB of disk space on your mirror for patches. Set up a separate volume and use a file system that can easily be extended, because you may need to increase this capacity over time.

   Best practice is to isolate Oracle Private Cloud Appliance ULN channels from other ULN channels. This eliminates the risk of downloading an enormous volume of unused packages, which would take many extra hours.

3. Register the host name of your local mirror in your local DNS.

   Configuring the upstream mirror on Oracle Private Cloud Appliance requires the fully qualified domain name; an IP address is not accepted.

4. Confirm that you have the appropriate version of `uln-yum-mirror` installed for your operating system.

   - Oracle Linux 8: version `0.4.0-5.el8` or later

     ```
     # yum --disablerepo=* --enablerepo=ol8_addons install uln-yum-mirror
     ```

   - Oracle Linux 7: version `0.3.0-10.el7` or later

     ```
     # yum --disablerepo=* --enablerepo=ol7_addons install uln-yum-mirror
     ```

5. Update the ULN mirror configuration by changing the `/etc/sysconfig/uln-yum-mirror` configuration file.

   - Specify that all versions of every available package must be mirrored. Set `ALL_PKGS=1`.
   - Prevent the mirror from synchronizing with ULN automatically. Set `CRON_ENABLED=0`. The mirror must be synchronized manually.

6. Enable updates for MySQL packages.

   These packages are signed with a different GPG key. The ULN mirror requires the public key to verify the package signatures and download the packages to the local mirror repository (`pca302_x86_64_mn`).

   a. Download the MySQL GPG key from https://repo.mysql.com/RPM-GPG-KEY-mysql-2022.

   b. Import the GPG key.

      ```
      # rpm --import RPM-GPG-KEY-mysql-2022
      ```

> **Note:**
>
> Additional information is provided in the Oracle Private Cloud Appliance Release Notes. See known issue "No Packages Available to Patch MySQL Cluster Database". It also contains steps to update an existing ULN mirror configuration that does not have the MySQL key yet.

Subscribe to the Private Cloud Appliance ULN Channels

Subscribe the ULN mirror server to the appropriate Private Cloud Appliance ULN channels.

> **Note:**
>
> ULN channels can be subscribed to once updates are added for download. If new channels are made available, repeat this procedure to subscribe to them, so the mirror can receive the updates.
>
> If no "PCA" channels appear, verify that your CSI is correctly enabled to display them.

> **Caution:**
>
> Only install patches from the "PCA" channels. Manually updating the appliance using other channels and other methods is not supported. Security and other updates to Oracle Linux will come through the "PCA" channels.

1. Log in to https://linux.oracle.com, go to the System Details page of the ULN mirror server, and click Manage Subscriptions.
2. Subscribe to the following ULN channels for Private Cloud Appliance:
   - PCA 3.0.2 Container Images
   - PCA 3.0.2 Firmware
   - PCA 3.0.2 Hypervisor
   - PCA 3.0.2 MN
   - PCA 3.0.2 OCI Compute Images
   - PCA 3.0.2 Region Registry

   The list also contains channels with source RPMs for the binary channels; their name contains "src". Do not subscribe to these "src" channels, because they are not used for patching and would take up significant space on your mirror.

   - PCA 3.0.2 Container Images src
   - PCA 3.0.2 Hypervisor src
   - PCA 3.0.2 MN src

In the Available channels list, scroll down to the "PCA 3.0.2" channels. Use the > button to move the channels in the preceding list to the Subscribed Channels column. Click the Save Subscriptions button.

3. Use the `yum repolist` command to verify that you have correctly subscribed to the Private Cloud Appliance channels.

   The example shows output from an Oracle Linux 7 mirror. The "status" column does not appear on Oracle Linux 8.

```
# sudo yum repolist
...
repo id                                repo name
status
pca302_x86_64_container_images         PCA 3.0.2 Container Images
492
pca302_x86_64_fw                       PCA 3.0.2
Firmware                           6
pca302_x86_64_hypervisor               PCA 3.0.2 Hypervisor
179
pca302_x86_64_mn                       PCA 3.0.2 MN
31
pca302_x86_64_oci                      PCA 3.0.2 OCI Compute
Images               5
pca302_x86_64_regionregistry           PCA 3.0.2 Region
Registry                 2
```

As an alternative to the ULN web interface, you can subscribe to ULN channels from the command line. See Oracle Linux: Managing ULN Channel Subscriptions via Command Line (Doc ID 1674425.1).

Populate the Package Repositories

During the local mirror setup, an `EngineeredSystems` directory is created. When packages are downloaded to the local mirror, they will be stored in subdirectories of `EngineeredSystems`. You must create soft links to these subdirectories to allow the system to locate the packages. Then you run the repository update script to download the packages to the respective local repositories.

> **Note:**
>
> As long as a channel contains no package updates, you cannot subscribe to it yet, which implies that its local mirror directory does not exist and no soft link can be created. Check your ULN registration regularly for new "PCA" channels. If new channels are found, repeat the procedure: subscribe to them, create the soft links, and run the repository update script.

> **Caution:**
>
> The repository update script (`uln-yum-mirror`) updates all mirrored channels, so use it carefully. You must keep the mirror repositories at a given version until all appliance components have been patched. Ensure that automatic synchronization is disabled. If the mirror is synchronized between patch operations, components will be patched to different software levels. This leads to unreliable appliance operation and potential service disruption.

1. Create soft links for your local mirror directories.

   The default location of the `EngineeredSystems` directory is `/var/www/html/yum/EngineeredSystems`. To ensure that the system can locate the packages in the subdirectories, create soft links in the parent directory `/var/www/html/yum` that point to the mirror subdirectories. The exact path to these directories differs between Oracle Linux 8 and Oracle Linux 7.

   - Soft links on an Oracle Linux 8 mirror:

     ```
     ln -s EngineeredSystems/pca302/hypervisor/x86_64/pca302_x86_64_hypervisor
     pca302_x86_64_hypervisor
     ln -s EngineeredSystems/pca302/containers/x86_64/
     pca302_x86_64_container_images pca302_x86_64_container_images
     ln -s EngineeredSystems/pca302/fw/x86_64/pca302_x86_64_fw pca302_x86_64_fw
     ln -s EngineeredSystems/pca302/mn/x86_64/pca302_x86_64_mn pca302_x86_64_mn
     ln -s EngineeredSystems/pca302/oci/x86_64/pca302_x86_64_oci pca302_x86_64_oci
     ln -s EngineeredSystems/pca302/regionregistry/x86_64/
     pca302_x86_64_regionregistry pca302_x86_64_regionregistry
     ```

   - Soft links on an Oracle Linux 7 mirror:

     ```
     ln -s EngineeredSystems/pca302/hypervisor/x86_64 pca302_x86_64_hypervisor
     ln -s EngineeredSystems/pca302/containers/x86_64 pca302_x86_64_container_images
     ln -s EngineeredSystems/pca302/fw/x86_64 pca302_x86_64_fw
     ln -s EngineeredSystems/pca302/mn/x86_64 pca302_x86_64_mn
     ln -s EngineeredSystems/pca302/oci/x86_64 pca302_x86_64_oci
     ln -s EngineeredSystems/pca302/regionregistry/x86_64
     pca302_x86_64_regionregistry
     ```

   > **Note:**
   >
   > This example assumes that all "PCA" channels contain updates. If a channel has no updates, you cannot subscribe to it, meaning the mirror directory and soft link cannot be created. Check your ULN registration whenever a new appliance software build is released, and repeat this procedure if new "PCA" channels are available.

2. Populate or update the repositories. The following command downloads all new packages to the local mirror file system. The initial download could take an hour or longer.

   ```
   # /usr/bin/uln-yum-mirror
   ```

> **Caution:**
>
> Keep the mirror repositories at a given version until all appliance components have been patched. Ensure that automatic synchronization is disabled. If the mirror is synchronized between patch operations, components will be patched to different software levels. This leads to unreliable appliance operation and potential service disruption.

3. Verify that a `repodata` directory was created at the location of a soft link.

   The presence of a `repodata` directory indicates that the repository was populated successfully. Perform this check for each new ULN channel you set up on the mirror. There is no need to repeat these checks when an existing local repository is updated.

   ```
   # ls -l /var/www/html/yum/pca302_x86_64_hypervisor/
   drwxr-xr-x. 2 root root 12288 Dec 19 03:20 getPackage
   drwxr-xr-x. 2 root root  4096 Dec 19 03:19 getPackageSource
   drwxr-xr-x. 2 root root  4096 Jan 15 03:41 repodata
   ```

Set the Appliance Upstream ULN Mirror

By design, compute nodes do not have access outside the appliance. Private Cloud Appliance uses the local mirror inside the data center as its upstream ULN mirror. An internal repository, located on the appliance internal shared storage, is synchronized with the upstream mirror to enable compute node access to the required packages.

Using the Service CLI

1. Configure the management node cluster to synchronize with the local ULN mirror and receive package updates. Set the *fully qualified domain name* of the data center mirror server using the `setUpstreamUlnMirror` command.

   > **Caution:**
   >
   > You must use the fully qualified domain name to reference the data center mirror server, not the system IP address.

   ```
   PCA-ADMIN> setUpstreamUlnMirror ulnMirrorLocation=http://host.example.com/yum
   Data:
     upstream channels are set UpstreamMirror status = success
   ```

2. To configure a more secure connection, see Using HTTPS to Reach the ULN Mirror Server.

3. To set up an additional channel for synchronization to the internal appliance repository, use the `addUpstreamUlnChannel` command. This adds the channel to the local yum repository configuration.

> **Note:**
>
> This step is required to set up the registry for the Oracle Container Engine for Kubernetes (OKE). However, it is automatically performed at the end of host patching.

```
PCA-ADMIN> addUpstreamUlnChannel ulnMirrorLocation=http://host.example.com/yum
channel=pca302_x86_64_regionregistry
```

To unconfigure a synchronized channel, use the `removeUpstreamUlnChannel` command.

Using the Service Web UI

> **Note:**
>
> This UI function is available in software version 3.0.2-b925538 and later.

Set the upstream ULN mirror as follows:

1. In the navigation menu, click Maintenance and select ULN Mirrors.
2. In the top-right corner of the ULN Mirrors page, click Set ULN Mirror.

   The ULN Mirror window appears.
3. Fill out the parameters:
   - **ULN Mirror:** the fully qualified domain name of the ULN mirror in your data center.
   - **Proxy:** If your data center uses a proxy server as an intermediary for Internet access, specify that server here.
4. Click Set ULN Mirror.

   The ULN mirror is set.

Using HTTPS to Reach the ULN Mirror Server

To connect to the ULN mirror using HTTPS, add the TLS trust information for the ULN mirror server to the appliance. The TLS trust information to add to the appliance must contain only a CA chain or an X.509 server certificate; the trust information on the appliance must not contain keys.

- If the server certificate is signed by a commercial Certificate Authority, do not add anything to the appliance. Skip this procedure.
- If the server certificate is signed by a non-commercial Certificate Authority, the TLS trust information to add to the appliance is the non-commercial CA chain file, in PEM or CRT format.
- If the server certificate is self-signed, the TLS trust information to add to the appliance is a copy of the server certificate, in PEM format.

Repeat this process whenever the X.509 server certificate on the ULN mirror server is replaced, such as when the certificate expires.

1. On the first management node, create the following directory if it does not already exist:

```
/etc/pca3.0/vault/customer_ca/
```

2. Copy the CA chain or X.509 server certificate to the `/etc/pca3.0/vault/customer_ca/` directory.

   If the ULN server certificate is not self-signed, copy the CA chain. If the ULN server certificate is self-signed (the Subject Key Identifier is the same as the Authority Key Identifier), copy the server certificate.

3. Run the following command:

   ```
   python3 /usr/lib/python3.6/site-packages/pca_foundation/secret_service/
   cert_generator/cert_generator_app.py -copy_to_mns
   ```

   The resulting TLS trust/certificate bundle is in the following directory on each management node:

   ```
   /etc/pca3.0/vault/certs/ca_outside_bundle.crt
   ```

3
Prepare for Patching

> **Caution:**
>
> The granular appliance architecture with built-in redundancy allows administrators to upgrade or patch components without downtime. However, resource capacity and performance might be reduced while an upgrade or patch workflow is in progress.
>
> We recommend that administrators responsible for upgrade or patching notify all Compute Enclave users in advance about such planned maintenance operations.
>
> This is particularly important for users of the Oracle Container Engine for Kubernetes (OKE), because new cluster deployments are not allowed during the maintenance window, and some types of application clusters might experience service interruptions.

Before you start a patching procedure, ensure that you have the correct permissions, that you refresh the local data center mirror and the secondary management node mirror, and that you have downloaded the RPM packages to the appropriate locations. You should also run health checks and perform a system backup before you begin patching procedures.

1. Verify you have permissions to perform patching operations. Log in to the Service Enclave with an administrator account and enter `showcustomcmds patchRequest` to ensure you have the correct permissions to use the patching commands.

   ```
   PCA-ADMIN> showcustomcmds patchRequest
       patchCN
       patchIlom
       patchOCIImages
       patchSwitch
       patchZfssa
       patchHost
       patchKubernetes
       patchVault
       patchPlatform
       patchMySQL
       patchEtcd
       setUpstreamUlnMirror
       syncUpstreamUlnMirror
       getUpstreamUlnChannels
       getUpstreamUlnChannel
       addUpstreamUlnChannel
       removeUpstreamUlnChannel
   ```

   Patching permissions are available to these groups: SuperAdmin, Admin, and DR Admin. For more information, see the Administrator Account Management chapter of the Oracle Private Cloud Appliance Administrator Guide.

2. Log in to the local mirror server and update the Oracle Private Cloud Appliance repositories by entering the following command:

```
# /usr/bin/uln-yum-mirror
```

> **Caution:**
>
> Keep the mirror repositories at a given version until all appliance components have been patched. Ensure that automatic synchronization is disabled.
>
> If the mirror is synchronized between patch operations, components will be patched to different software levels. This leads to unreliable appliance operation and potential service disruption.

3. Ensure the patch RPM files are updated and in the location you expect, and note the path.
4. After you have updated the local mirror server, update the local repository used for compute node patches by running the `syncUpstreamUlnMirror` command:

```
PCA-ADMIN> syncUpstreamUlnMirror
Command: syncUpstreamUlnMirror
Status: Success
Time: 2022-01-04 15:52:07,120 UTC
Data:
  Upstream mirror sync started. UpstreamMirror status = success
```

> **Note:**
>
> Alternatively, you can synchronize the appliance local repository from the Service Web UI. In the navigation menu, go to Maintenance and select ULN Mirrors. Click Sync ULN Mirror. However, this UI function is unavailable in software version 3.0.2-b892153.

> **Caution:**
>
> ULN channels only become available for subscription, and thus mirroring, when they contain updates. When a new channel is announced, you must repeat a part of the process described in Configure Your Environment for Patching: subscribe to the new channel, create the appropriate soft link, download packages to the mirror, and if necessary, sync the appliance internal repository.

5. View the status of the local repository update and confirm it completes successfully.

   Alternatively, you can perform this step in the Service Web UI.

   - With appliance software 3.0.2-b852928 or earlier, use the `showUpstreamUlnMirror` command.

```
PCA-ADMIN> showUpstreamUlnMirror
Command: showUpstreamUlnMirror
Status: Success
Time: 2022-01-24 17:29:48,965 UTC
```

```
Data:
  Mirror URI = https://host.example.com/yum
```

- With appliance software newer than version 3.0.2-b892153 use the `getUpstreamUlnChannels` command. For more details, display the channel properties.

```
PCA-ADMIN> getUpstreamUlnChannels
Data:
  id                          lastSync             syncStatus
  --                          --------             ----------
  pca302_x86_64_hypervisor    2023-06-22/09:46:01   success
  pca302_x86_64_mn            2023-06-22/09:46:04   success

PCA-ADMIN> getUpstreamUlnChannel channel=pca302_x86_64_hypervisor
Data:
  Type = UlnChannel
  Channel Name = pca302_x86_64_hypervisor
  Last Synced = 2023-06-22/09:46:01
  Sync Status = success
  Message = upstream channel sync succeeded
  Mirror URI = http://host.example.com/yum/pca302_x86_64_hypervisor
```

> **Note:**
>
> In version 3.0.2-b892153 the command is `getUpstreamUlnMirror(s)`. It is functionally identical to `getUpstreamUlnChannel(s)`.

6. Before starting any patching activities, create backups of these critical components: the MySQL database, the ZFS Storage Appliance and the Secret Service (Vault).

   a. Start the three backup operations.

```
PCA-ADMIN> backup target=vault
PCA-ADMIN> backup target=zfs
PCA-ADMIN> backup target=mysql
```

   b. Use the backup job ID to check the status of the backups. Make sure they have completed successfully before you proceed to the next step.

```
PCA-ADMIN> getBackupJobs
[...]

id
                displayName                components

--
                -----------                ----------
  ocid1.brs-
job.PCA3X62D9C1.mypca.iew5tphpgr3h6mhliw2fai2ywvv386a0xc7isfo8kisj0wrcx114irnit
6ot   brs-job-1668419850-backup   mysql
  ocid1.brs-
job.PCA3X62D9C1.mypca.9oaeaa2kw5crqfcjkh8kyhbxcv8bwh0f4ud6n3lucf802oj15ss3k3987
4bc   brs-job-1668419842-backup   zfs
  ocid1.brs-
job.PCA3X62D9C1.mypca.joopwuv9403uzbfrh4x9mprmoduh3ljais6ex233v1b21ccqywu4a3vqy
kgm   brs-job-1668419778-backup   vault

PCA-ADMIN> getBackupJob backupJobId=ocid1.brs-
job.PCA3X62D9C1.mypca.iew5tphpgr3h6mhliw2fai2ywvv386a0xc7isfo8kisj0wrcx114irnit
```

```
6ot
[...]
  Status = success
  Components = mysql
```

See the Backup and Restore chapter of the Oracle Private Cloud Appliance Administrator Guide.

Update the Local Packages Using the Service Web UI

1. In the navigation menu, click ULN Mirror.
2. In the top-right corner of the ULN Mirror page, click Update ULN Mirror.

   The ULN Mirror window appears.
3. Click Sync ULN Mirror.

   The ULN mirror is updated.
4. Ensure the system is in a ready state for patching, as described in the Service CLI steps above.
   a. Create backups of the critical system components: MySQL database, ZFS Storage Appliance, Secret Service (Vault).
   b. Prepare the patching environment to ensure that the latest patching functionality, RPM packages, and YUM configuration are in place.

Prepare the Patching Environment

Patching operations rely on the same logic as upgrades, but use RPM packages provided through ULN channels instead of an ISO image. Before you start patching an Private Cloud Appliance you must ensure that all updated RPM packages are available in the appropriate locations, and that the Upgrader packages in use on the system are up-to-date.

No patching operations will be allowed to run if the system detects that the Upgrader is not the latest available version.

> **Caution:**
>
> To prevent inconsistencies while executing the upgrade plan later on, it is critical that both parts of the preparation process, Upgrade PreConfig and preUpgrade, are completed together in the specified order. If at any time you need to rerun the preUpgrade command, you must rerun the preceding command first.

Prerequisite Version
The latest Upgrader code automatically enforces prerequisite software versions on your Private Cloud Appliance. In the early stages of upgrade or patch preparation, the Upgrader service validates the currently installed appliance software version against the new target version. The preparation process (`upgradePreConfig`) documented in this section will only proceed if validation is successful.

If the appliance is not running at least the minimum required version, the Upgrader exits the process and rolls back the environment to its previous state. View the details of the failed upgrade job:

```
PCA-ADMIN> getupgradejob upgradeJobId=1700153626051-prepare-40046
Data:
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_prepare_environment_2023_11_16-16.53.46.log
  Arguments = [...]
  Status = Failed
[...]
  Tasks 23 - Name = Check Prerequisite Build Version
  Tasks 23 - Description = Check current build version not lower than prerequisite
version
  Tasks 24 - Name = Check Prerequisite Build Version
  Tasks 24 - Message = (("Caught exception while checking prerequisite build number
  Exception: Command: ['/usr/bin/python3', '/var/lib/pca-upgrader/
prerequisite_build_validator.py',
  'rack=PCA', 'upgrade=ISO'] failed (1): stderr: b'' stdout: b'PCA version is lower
than prerequisite build,
  must upgrade to prerequisite build 3.0.2-b799577 to proceed further upgrade\\n'",),
{})
  Tasks 24 - Status = Failed
```

You must first install the prerequisite version, which is indicated by the error message in the upgrade job output.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade PreConfig.
4. Fill out the upgrade request parameters:
   - **Option:** Enter *ULN*.
   - **Location:** Enter the path to the ULN mirror in the data center, using its fully qualified domain name.
   - **ISO Checksum:** This parameter applies to upgrades from an ISO image and can be ignored.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Alternative ULN Channel:** This parameter forces the request to use a non-standard ULN channel. Do not use this option unless Oracle explicitly instructs you to do so.
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table. Package repositories are populated with the latest packages and the YUM configuration is updated as needed.
6. When the previous upgrade job has completed successfully, create another upgrade request.

7. Choose *Upgrade* as the Request Type again, but select a different upgrade request type: preUpgrade.
8. Fill out the upgrade request parameters:
   - **Action:** Enter *start* to retrieve the new version of the upgrader. (To check the status of the previous preUpgrade job, enter *status* instead.)
   - **Type:** Enter *ULN*. The upgrader packages are picked from the appropriate mirrored ULN channel.
9. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table. When the job has completed successfully, the upgrader is up-to-date and ready for use.

Using the Service CLI

1. Populate the repositories with the latest packages and apply the latest YUM configuration. Use the pre-configuration command as shown below. Check that the job completes successfully.

```
upgradePreConfig option=ULN location=https://host.example.com/yum
[...]
Data:
  Service request has been submitted. Upgrade Job Id = 1692849609034-
prepare-45676 Upgrade Request Id = UWS-edfa3b32-c32a-4b67-8df5-2357096052bf

PCA-ADMIN> getUpgradeJob upgradeJobId=1692849609034-prepare-45676
[...]
  Status = Passed
  Execution Time(sec) = 616
```

2. Enable the latest patching functionality using the pre-upgrade command. Check that the job completes successfully.

```
PCA-ADMIN> preUpgrade action=start type=ULN

PCA-ADMIN> preUpgrade action=status
[...]
Data:
   The previous pre-upgrade task succeeded!
   Pre upgrade status = SUCCESS
```

Checking Current Version of Components

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

To evaluate patching requirements and the impact of the patch operations you can view the current state of the top-level rack components. The system lists the most important build and version numbers of all the components for which a patch procedure is documented in this guide.

The component version list can be viewed in two ways:

- In the Service Web UI, go to Maintenance in the navigation menu and select Component Version.
- In the Service CLI, enter the following command:

```
PCA-ADMIN> getComponentVersions
Data:
  id                        component    iso              version
  --                        ---------    ---              -------
  100.96.2.64               compute      3.0.2-b1046481   3.0.2-687
  100.96.2.65               compute      3.0.2-b1046481   3.0.2-687
  100.96.2.66               compute      3.0.2-b1046481   3.0.2-687
  100.96.2.67               compute      3.0.2-b1046481   3.0.2-687
  100.96.2.68               compute      3.0.2-b1046481   3.0.2-687
  generic                   etcd         3.0.2-b1049367   3.3.10
  100.96.2.33               host         3.0.2-b1049367   oraclelinux-
release-7.9-1.0.9
  100.96.2.34               host         3.0.2-b1049367   oraclelinux-
release-7.9-1.0.9
  100.96.2.35               host         3.0.2-b1049367   oraclelinux-
release-7.9-1.0.9
  100.96.0.33               ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.34               ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.35               ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.64               ilom         3.0.2-b1049367   5.1.2.20.a
  100.96.0.65               ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.66               ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.67               ilom         3.0.2-b1049367   5.1.2.20.a
  100.96.0.68               ilom         3.0.2-b1049367   5.1.1.21
  generic                   kubernetes   3.0.2-b1049367   1.25.15-1
  generic                   mysql        3.0.2-b1049367   8.0.33-1.1
  Oracle-Linux-7.9          ociImages    3.0.2-b1049367   2023.09.26_0
  Oracle-Linux-8            ociImages    3.0.2-b1049367   2023.09.26_0
  Oracle-Linux-9            ociImages    3.0.2-b1049367   2023.09.26_0
  Oracle-Linux8-OKE-1.26.6  ociImages    3.0.2-b1049367   20240210
  Oracle-Linux8-OKE-1.27.7  ociImages    3.0.2-b1049367   20240209
  Oracle-Linux8-OKE-1.28.3  ociImages    3.0.2-b1049367   20240210
  Oracle-Solaris-11         ociImages    3.0.2-b1049367   2023.10.16_0
  generic                   platform     3.0.2-b1046481   None
  leaf                      switch       3.0.2-b1049367   10.3.4a
  mgmt                      switch       3.0.2-b1049367   10.3.4a
  spine                     switch       3.0.2-b1049367   10.3.4a
  generic                   vault        3.0.2-b1049367   v1.7.1-3
  generic                   zfssa        3.0.2-b1046481
2013.06.05.8.57.1-2.57.5501.1
```

4
Checking Upgrade Plan Status and Progress

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

During the execution of the pre-upgrade command, an *upgrade plan* is also generated. The plan is based on a comparison of the currently installed components on the rack, and the target component versions and latest packages downloaded to shared storage during the preparation of the upgrade environment. The resulting upgrade plan shows for which components a patching procedure needs to be executed in the next phase.

All components except firmware must be patched in a prescribed order. The upgrade plan will prevent a component patch procedure from starting if the preceding patch operations have not been completed. An error message informs the administrator which components need to be patched first.

> **Note:**
>
> In general, firmware may be patched whenever new versions are made available for your system. Firmware patches can be applied in no particular order and independently of other components.
>
> However, there is an extra requirement: when patching to appliance software version 3.0.2-b1081557 or later, the ZFS Storage Appliance firmware must be patched before all other components.

This is the order of operations enforced through the upgrade plan:

1. Prepare upgrade environment (Upgrade PreConfig)
2. Upgrade the Upgrader (PreUpgrade)
3. ZFS Storage Appliance firmware (version 3.0.2-b1081557 or later)
4. Compute nodes
5. Host operating system of management nodes
6. MySQL cluster database
7. Secret service (including Etcd and Vault)
8. Kubernetes container orchestration packages (platform layer)
9. Containerized microservices
10. Oracle Cloud Infrastructure images

Once the patching environment has been prepared, all patch operations required to bring the system up-to-date are listed in the upgrade plan. Whenever a patch procedure has been completed successfully, the upgrade plan is updated with the latest status: for patched components the source and target versions are identical and the "upgrade required" flag is disabled.

At any point in time you can check how far the system has progressed through the upgrade plan. It indicates which components are already up-to-date and which still require patching.

The upgrade plan can be viewed in two ways:

- In the Service Web UI, go to Maintenance in the navigation menu and select Upgrade Plan.
- In the Service CLI, enter the following command:

```
PCA-ADMIN> getUpgradePlan
Data:
  id                          component    currentBuild     targetBuild
currentVersion                   targetVersion
requireReboot   timeEstimation (min)   requireUpgrade   impactedInfra
  --                          ---------    ------------     -----------
--------------                   -------------
-------------   --------------------   --------------   -------------
  generic                     zfssa        3.0.2-b1053709   3.0.2-b1053709
2013.06.05.8.57.1-2.57.5501.4   2013.06.05.8.57.1-2.57.5501.4
false           45                      false            host,compute
  100.96.2.64                 compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                       3.0.2-691
false           20                      false            compute
  100.96.2.65                 compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                       3.0.2-691
false           20                      false            compute
  100.96.2.66                 compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                       3.0.2-691
false           20                      false            compute
  100.96.2.67                 compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                       3.0.2-691
false           20                      false            compute
  100.96.2.68                 compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                       3.0.2-691
false           20                      false            compute
  100.96.2.33                 host         3.0.2-b1053709   3.0.2-b1053709
oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
false           35                      false            host
  100.96.2.34                 host         3.0.2-b1053709   3.0.2-b1053709
oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
false           35                      false            host
  100.96.2.35                 host         3.0.2-b1053709   3.0.2-b1053709
oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
false           35                      false            host
  generic                     mysql        3.0.2-b1053709   3.0.2-b1053709
8.0.33-1.1                      8.0.33-1.1
false           15                      false            host
  generic                     etcd         3.0.2-b1053709   3.0.2-b1053709
3.3.10                          3.3.10
false           5                       false            host
  generic                     vault        3.0.2-b1053709   3.0.2-b1053709
v1.7.1-3                        v1.7.1-3
false           5                       false            host
```

```
  generic                    kubernetes   3.0.2-b1053709   3.0.2-b1053709
1.25.7-1                        1.25.7-1                         false
80                      false            host,compute
  generic                    platform     3.0.2-b1053709   3.0.2-b1053709
None                            None                             false
50                      false            host,compute
  Oracle-Linux-7.9           ociImages    3.0.2-b1053709   3.0.2-b1053709
2023.09.26_0                    2023.09.26_0                     false
5                       false            host
  Oracle-Linux-8             ociImages    3.0.2-b1053709   3.0.2-b1053709
2023.09.26_0                    2023.09.26_0                     false
5                       false            host
  Oracle-Linux-9             ociImages    3.0.2-b1053709   3.0.2-b1053709
2023.09.26_0                    2023.09.26_0                     false
5                       false            host
  Oracle-Linux8-OKE-1.26.6   ociImages    3.0.2-b1053709   3.0.2-b1053709
20240210                        20240210                         false
5                       false            host
  Oracle-Linux8-OKE-1.27.7   ociImages    3.0.2-b1053709   3.0.2-b1053709
20240209                        20240209                         false
5                       false            host
  Oracle-Linux8-OKE-1.28.3   ociImages    3.0.2-b1053709   3.0.2-b1053709
20240210                        20240210                         false
5                       false            host
  Oracle-Solaris-11          ociImages    3.0.2-b1053709   3.0.2-b1053709
2023.10.16_0                    2023.10.16_0                     false
5                       false            host
  100.96.0.33                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.1.21                        5.1.1.21                         false
10                      false            host,compute
  100.96.0.34                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.1.21                        5.1.1.21                         false
10                      false            host,compute
  100.96.0.35                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.1.21                        5.1.1.21                         false
10                      false            host,compute
  100.96.0.64                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.2.20.a                      5.1.2.20.a                       false
10                      false            host,compute
  100.96.0.65                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.1.21                        5.1.1.21                         false
10                      false            host,compute
  100.96.0.66                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.1.21                        5.1.1.21                         false
10                      false            host,compute
  100.96.0.67                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.2.20.a                      5.1.2.20.a                       false
10                      false            host,compute
  100.96.0.68                ilom         3.0.2-b1053709   3.0.2-b1053709
5.1.1.21                        5.1.1.21                         false
10                      false            host,compute
  leaf                       switch       3.0.2-b926028    3.0.2-b1053709
10.2.3                          10.3.4a                          false
60                      true             host,compute
  mgmt                       switch       3.0.2-b926028    3.0.2-b1053709
10.2.3                          10.3.4a                          false
60                      true             host,compute
  spine                      switch       3.0.2-b926028    3.0.2-b1053709
10.2.3                          10.3.4a                          false
60                      true             host,compute
```

5
Checking System Upgrade History

Information about all component upgrades and patches is stored in upgrade jobs, which can be consulted in the Service Web UI and Service CLI. Over time, as the system goes through multiple upgrades, the large number of entries might make the list difficult to interpret. The upgrade history provides a clear way to drill down into the details of the upgrade and patching activity on your appliance.

The upgrade history presents the information from all upgrade and patch jobs in a categorized way so you can see which version upgrades have been performed, which jobs have been run for each of those upgrades, and from which source (ISO upgrade or ULN patch). Details include build versions, component versions before and after, job completion, success or failure, time stamps, and duration.

Appliance software builds are installed onto the appliance either through upgrade from an ISO image or patching from ULN. Display the build history by running the following command.

```
PCA-ADMIN> GetUpgradeHistory
Data:
  id                                    From Build       To Build          Type
Status         Start Time               End Time               Actual Upgrade
Time(min)    Total Upgrade Time(min)
  --                                    -----            -------           -----
--------       ----------               --------
--------------             ------------
  pca-upgrade-history-3.0.2-b951413    3.0.2-b868711    3.0.2-b951413     ISO
Incomplete     2023-06-14T10:32:14      2023-06-14T15:32:14
300                        600
  pca-upgrade-history-3.0.2-b868711    3.0.2-b854356    3.0.2-b868711     ULN
Completed      2023-01-01T15:10:07      2023-01-01T19:10:07
250                        500
  pca-upgrade-history-3.0.2-b854356    3.0.2-b790137    3.0.2-b854356     ISO
Completed      2022-12-14T06:01:00      2022-12-14T12:01:00
350                        700
```

If necessary, you can filter results by type.

```
PCA-ADMIN> GetUpgradeHistory type=uln
Data:
  id                                    From Build       To Build          Type
Status         Start Time               End Time               Actual Upgrade
Time(min)    Total Upgrade Time(min)
  --                                    -----            -------           -----
--------       ----------               --------
--------------             ------------
  pca-upgrade-history-3.0.2-b868711    3.0.2-b854356    3.0.2-b868711     ULN
Completed      2023-01-01T15:10:07      2023-01-01T19:10:07
250                        500
```

To display the upgrade job list related to a particular build, copy the build ID and run this command:

```
PCA-ADMIN> GetUpgradeHistoryDetails id=pca-upgrade-history-3.0.2-b951413
Data:
```

```
  id           component  Timestamp             From Version
To Version                      Status   Job ID
Time Taken(min)
  --           ---------  ---------             ------------
----------                      -------  -------
----------
  spine        cisco      2023-06-14T10:32:14   9.3.2
10.2.3                          Failed   1686766376945-cisco-1509
50
  mgmt         cisco      2023-06-14T10:22:14   9.3.2
10.2.3                          Passed   1686762287214-cisco-31252
60
  leaf         cisco      2023-06-14T10:12:14   9.3.2
10.2.3                          Passed   1686758831077-cisco-35671
45
  generic      zfssa      2023-06-14T09:52:14   2013.06.05.8.40.1-2.40.4958.31
2013.06.05.8.48.1-2.48.5222.1   Passed   1686768921264-zfssa-19292
30
  100.96.0.66  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19293
55
  100.96.0.65  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19292
50
  100.96.0.64  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19294
60
  100.96.0.35  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19295
35
  100.96.0.34  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19296
30
  100.96.0.33  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19297
65
  generic      platform   2023-06-14T09:12:14   None
None                            Passed   1686552138506-platform-48794
100
  generic      kubernetes 2023-06-14T09:12:14   1.20.6-1
1.25.7-1                        Passed   1685379294807-kubernetes-72808
40
  generic      vault      2023-06-14T09:12:14   v1.7.1-3
v1.7.1-3                        Passed   1685614045193-vault-47652
63
  generic      etcd       2023-06-14T09:02:14   3.3.10
3.3.10                          Passed   1685613743232-etcd-83924
50
  generic      mysql      2023-06-14T08:52:14   8.0.28-1.1
8.0.30-1.1                      Passed   1685378389035-mysql-90009
45
  100.96.2.35  host       2023-06-14T08:42:14   oraclelinux-release-7.9-1.0.9
oraclelinux-release-7.9-1.0.9   Passed   1686726229799-host-68919
20
  100.96.2.34  host       2023-06-14T08:32:14   oraclelinux-release-7.9-1.0.9
oraclelinux-release-7.9-1.0.9   Passed   1686726229799-host-68919
30
  100.96.2.33  host       2023-06-14T08:22:14   oraclelinux-release-7.9-1.0.9
oraclelinux-release-7.9-1.0.9   Passed   1686726229799-host-68919
25
  100.96.2.66  compute    2023-06-14T07:52:14   3.0.2-502
```

```
3.0.2-630                       Passed   1685607717395-compute-53331          50
  100.96.2.65  compute    2023-06-14T07:42:14   3.0.2-502
3.0.2-630                       Passed   1685606912300-compute-91673          66
  100.96.2.64  compute    2023-06-14T07:32:14   3.0.2-502
3.0.2-630                       Passed   1685372050358-compute-50568          30
  100.96.2.64  compute    2023-06-14T07:22:14   3.0.2-502
3.0.2-630                       Failed   1685372050358-compute-50560          45
  generic      preupgrade 2023-06-14T06:32:14   3.0.2-b868711                3.0.2-
b951413                    N/A      N/A                                 N/A
```

6
Patching Individual Components

The granular patching mechanism allows you to perform patching procedures for individual hardware and software components. Besides the components included in the management node patch, you can also patch different categories of firmware, the operating system and appliance-specific software on the compute nodes, and Oracle Cloud Infrastructure images.

All components must be patched in a prescribed order. The upgrade plan will prevent a component patch procedure from starting if the preceding patch operations have not been completed.

> **Note:**
>
> In general, firmware may be patched whenever new versions are made available for your system. Firmware patches can be applied in no particular order and independently of other components.
>
> However, there is an extra requirement: when patching to appliance software version 3.0.2-b1081557 or later, the ZFS Storage Appliance firmware must be patched before all other components.

This is the order of operations enforced through the upgrade plan:

1. Prepare upgrade environment (Upgrade PreConfig and PreUpgrade)
2. ZFS Storage Appliance firmware (version 3.0.2-b1081557 or later)
3. Compute nodes
4. Management nodes
5. MySQL cluster database
6. Etcd
7. Vault
8. Kubernetes cluster
9. Platform (containerized microservices)

Every patch operation is preceded by a set of pre-checks. These are built into the code and will report an error if the system is not in the required state for patching. Patching will only begin if all pre-checks are passed.

You can use the pre-checks to test in advance for any system health issues that would prevent a successful patch. After preparing the environment for patching, run any or all of the patch commands with the "verify only" option. In the Service Web UI this option is activated with a check box when you create the patch request; in the Service CLI you use the optional patch command parameter shown in this example:

```
PCA-ADMIN> patchKubernetes ULN=http://host.example.com/yum verifyOnly=True
[...]
```

```
PCA-ADMIN> getUpgradeJobs
  id
upgradeRequestId                              commandName          result
  --
----------------                              -----------          ------
  1632849609034-kubernetes_verify-10575   UWS-8995e5b7-a237-4717-
bb5c-01f1cf85daf0    kubernetes_verify    Passed
```

If issues are detected, you can resolve them before the planned patch window, and keep the actual patching operations as fluent and short as possible.

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

Patching a Compute Node

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.
>
> When patching to appliance software version 3.0.2-b1081557 or later, the ZFS Storage Appliance firmware must be patched before all other components. For more information, see Checking Upgrade Plan Status and Progress.

The compute node patching ensures that the latest Oracle Linux kernel and user space packages are installed, as well as the `ovm-agent` package with appliance-specific optimizations. Compute nodes must be provisioned and locked, then patched one at a time, concurrent patches are not supported. After a successful patch, when a compute node has rebooted, the administrator must manually remove the locks to allow the node to return to normal operation.

Ensure synchronization of the mirror on the shared storage is complete prior to compute node patching by issuing the `syncUpstreamUlnMirror` command. For more information, see Prepare for Patching.

> **Note:**
>
> In case the ILOM also needs to be patched, you can integrate it into this procedure by executing the optional steps. The combined procedure eliminates the need to evacuate and reboot the same node twice.

> **Note:**
>
> In software versions 3.0.2-b892153 and later all patch operations are based on the upgrade plan, which is generated when the pre-upgrade command is executed. For more information, see Prepare for Patching. When a component is already at the required version, the patch operation is skipped. However, patching with the same version can be forced using the Service Web UI or Service CLI command option (`force=True`), if necessary.

Using the Service Web UI

1. Set the provisioning and maintenance locks for the compute node you are about to patch. Ensure that no active compute instances are present on the node.

   > **Caution:**
   >
   > Depending on the high-availability configuration of the Compute service, automatic instance migrations can prevent you from successfully locking a compute node. For more information, refer to the following sections in the Hardware Administration chapter of the Oracle Private Cloud Appliance Administrator Guide:
   >
   > - Migrating instances and locking a compute node: see "Performing Compute Node Operations".
   > - Compute service HA configuration: see "Configuring the Compute Service for High Availability".

   a. In the navigation menu, click Rack Units. In the Rack Units table, click the name of the compute node you want to patch to display its detail page.

   b. In the top-right corner of the compute node detail page, click Controls and select the Provisioning Lock command.

   c. When the provisioning lock has been set, click Controls again and select the Maintenance Lock command. This command might fail if instance migrations are in progress. Wait a few minutes and retry.

2. In the navigation menu, click Upgrade & Patching.

3. Optionally, patch the server ILOM first.

   a. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch. The Create Request window appears.

   b. Choose *Patch* as the Request Type. Select the appropriate patch request type: Patch ILOM.

      Fill out the server's assigned IP address in the ILOM network. This is an IP address in the internal 100.96.0.0/23 range.

   c. Click Create Request. The new patch request appears in the Upgrade Jobs table.

   d. Wait 5 minutes to allow the ILOM patch job to complete. Then proceed to patching the host.

4. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.

5. Select the appropriate patch request type: Patch CN.

6. If required, fill out the request parameters:

   - **Host IP:** Enter the compute node's assigned IP address in the internal administration network. This is an IP address in the internal 100.96.2.0/23 range.
   - **ULN:** Enter the fully qualified domain name of the ULN mirror in your data center. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
   - **Advanced Options JSON:** Not available.

7. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.

8. When the compute node has been patched successfully, release the provisioning and maintenance locks.

   For more information, refer to "Performing Compute Node Operations" in Hardware Administration.

   a. Open the compute node detail page.

   b. In the top-right corner of the compute node detail page, click Controls and select the Maintenance Unlock command.

   c. When the maintenance lock has been released, click Controls again and select the Provisioning Unlock command.

Using the Service CLI

1. Get the IP address of the compute node you intend to patch.

2. Set the provisioning and maintenance locks for the compute node you are about to patch.

> **Caution:**
>
> Depending on the high-availability configuration of the Compute service, automatic instance migrations can prevent you from successfully locking a compute node. For more information, refer to the following sections in the Hardware Administration chapter of the Oracle Private Cloud Appliance Administrator Guide:
>
> - Migrating instances and locking a compute node: see "Performing Compute Node Operations".
> - Compute service HA configuration: see "Configuring the Compute Service for High Availability".

```
PCA-ADMIN> list ComputeNode
Data:
```

```
  id                                     name       provisioningState
provisioningType
  --                                     ----       -----------------
----------------
  363a26f4-fa34-4e4c-8e17-a1671a0b77d1   pcacn001   Provisioned         KVM
  9e8745c7-52e3-4aae-984c-e198869ee2cc   pcacn002   Provisioned         KVM
  56a9ecda-2402-427f-92d1-7f9be57dba36   pcacn003   Provisioned         KVM

PCA-ADMIN> provisioningLock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
PCA-ADMIN> maintenanceLock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
```

3. Optionally, patch the server ILOM first.

   a. Enter the ILOM patch command.
   Syntax (entered on a single line):

   ```
   patchIlom
   hostIp=<ilom-ip>
   ```

   Example:

   ```
   PCA-ADMIN> upgradeIlom hostIp=100.96.0.64
   Data:
     Service request has been submitted. Upgrade Job Id = 1620921089806-
   ilom-21480 Upgrade Request Id = UWS-732d6fce-9f06-4329-b972-d093bee40010

   PCA-ADMIN> getUpgradeJob upgradeJobId=1620921089806-ilom-21480
   ```

   b. Wait 5 minutes to allow the ILOM patch job to complete. Then proceed to patching the host.

4. Enter the compute node patch command.

   Syntax (entered on a single line):

   ```
   patchCN
   hostIp=<compute-node-ip>
   [optional] uln=<http|https>://<hostname.domainname>/<sub-directories>
   ```

   The parameter marked optional is deprecated in software version 3.0.2-b892153 and later. For earlier versions, include the fully qualified domain name of the ULN mirror with the command.

   Example:

   ```
   PCA-ADMIN> patchCN hostIp=100.96.2.64 ULN=http://host.example.com/yum
   Command: patchCN hostIp=100.96.2.64 ULN=http://host.example.com/yum
   Status: Success
   Time: 2023-01-01 21:06:56.849 UTC
   Data: Service request has been submitted.  Upgrade Job ID = 1685372050358-
   compute-50568 \
   Upgrade Request ID = UWS-f226d7d2-549d-4902-8614-e1f40bdc9ff6
   ```

5. Use the request ID and the job ID to check the status of the patching process.

   ```
   PCA-ADMIN> getUpgradeJobs
   Command: getUpgradeJobs
   Status: Success
   Time: 2023-01-01 21:09:34.745 UTC
   Data:
     id                              upgradeRequestId
   commandName   result
     --                              ----------------
   -----------   ------
     1685372050358-compute-50568     UWS-f226d7d2-549d-4902-8614-e1f40bdc9ff6
   ```

```
compute       Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1685372050358-compute-50568
Command: getUpgradeJob upgradeJobId=1685372050358-compute-50568
Status: Success
Time: 2023-01-01 21:10:13,804 UTC
Data:
  Upgrade Request Id = UWS-f226d7d2-549d-4902-8614-e1f40bdc9ff6
  Name = compute
[...]
```

6. When the compute node patch has completed successfully and the node has rebooted, release the locks.

   For more information, refer to "Performing Compute Node Operations" in the Hardware Administration section of the Oracle Private Cloud Appliance Administrator Guide.

   ```
   PCA-ADMIN> maintenanceUnlock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
   PCA-ADMIN> provisioningUnlock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
   ```

7. Proceed to the next compute node and repeat this procedure.

Patching the Management Node Operating System with Appliance Software 3.0.2-b852928 or Earlier

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

The Oracle Linux host operating system of the management nodes must be patched one node at a time; a rolling patch of all management nodes is not possible. This patching process, which involves updating the kernel and system packages, must always be initiated from the management node that holds the cluster virtual IP. Thus, in a three-management-node cluster, when you have patched two management nodes, you must reassign the cluster virtual IP to one of the patched management nodes and execute the final patch command from that node. Each management node must be rebooted after a patch is applied.

You must patch management nodes one at a time, using each one's internal IP address as a command parameter. To obtain the host IP addresses, use the Service CLI command `show ManagementNode name=<pcamn01>` and look for the `Ip Address` in the output.

You cannot complete all of the patching tasks required in the Service Web UI for this component. Use the Service CLI to patch the management nodes.

Using the Service CLI

1. Get the IP address of the management node for which you intend to upgrade the host operating system.
2. Run the Service CLI from the management node that holds the management cluster virtual IP.

a. Log on to one of the management nodes and check the status of the cluster.

```
# ssh root@pcamn01
# pcs status
Cluster name: mncluster
Stack: corosync
Current DC: pcamn02 (version 1.1.23-1.0.1.el7-9acf116022) - partition with
quorum

Online: [ pcamn01 pcamn02 pcamn03 ]

Full list of resources:

 scsi_fencing         (stonith:fence_scsi):          Stopped (disabled)
 Resource Group: mgmt-rg
     vip-mgmt-int     (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-host    (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-ilom    (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-lb      (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-ext     (ocf::heartbeat:IPaddr2):      Started    pcamn02
     l1api            (systemd:l1api):               Started    pcamn02
     haproxy          (ocf::heartbeat:haproxy):      Started    pcamn02
     pca-node-state   (systemd:pca_node_state):      Started    pcamn02
     dhcp             (ocf::heartbeat:dhcpd):        Started    pcamn02
     hw-monitor       (systemd:hw_monitor):          Started    pcamn02

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

In this example, the command output indicates that the node with host name `pcamn02` currently holds the cluster virtual IP.

3. Log in to the management node virtual IP and launch the Service CLI.

```
# ssh -l admin 100.96.2.32 -p 30006
```

4. Enter the patch command.

Choose one of the management nodes that is not currently hosting the virtual IP. In the prior example, `pcamn02` holds the cluster virtual IP, so choose either `pcamn01` or `pcamn03` as your patch target.

Syntax (entered on a single line):

```
patchHost
hostIp=<management-node-ip>
```

Example:

```
PCA-ADMIN> patchHost hostIp=100.96.2.33
Command: patchHost hostIp=100.96.2.33
Status: Success
Time: 2022-01-01 21:06:56.849 UTC
Data: Service request has been submitted.  Upgrade Job ID = 1632990827394-
host-56156 \
Upgrade Request ID = UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
```

5. Use the job ID to check the status of the patch process. The job ID is listed in the output of the patch command.

```
PCA-ADMIN> getUpgradeJob upgradeJobId=1632990827394-host-56156
```

At the end of the patching process, the management node is rebooted automatically. Wait approximately 5 minutes until the management node restarts.

6. When the first management node host operating system patch has completed successfully, execute the same command for the next management node.

```
PCA-ADMIN> patchHost hostIp=100.96.2.35
```

At the end of the patching process, the management node is rebooted automatically. Wait approximately 5 minutes until the management node restarts.

7. When the second management node host operating system patch has completed successfully, move the cluster virtual IP to one of the upgraded management nodes.

```
# ssh root@pcamn01
root@pcamn01's password:
Last login: Mon Jan 10 20:50:28 2022
# pcs resource move mgmt-rg pcamn01
# pcs status
Cluster name: mncluster
Stack: corosync
[...]
 scsi_fencing   (stonith:fence_scsi):   Stopped (disabled)
 Resource Group: mgmt-rg
     vip-mgmt-int       (ocf::heartbeat:IPaddr2):       Started pcamn01
     vip-mgmt-host      (ocf::heartbeat:IPaddr2):       Started pcamn01
[...]
```

Moving the cluster virtual IP to another management node should only take a number of seconds and will close your current connection.

8. Log in to the management node virtual IP and launch the Service CLI to execute the host operating system patch for the final management node.

```
# ssh -l admin 100.96.2.32 -p 30006
PCA-ADMIN> patchHost hostIp=100.96.2.34
```

At the end of the patching process, the management node is rebooted automatically. Wait approximately 5 minutes until the management node restarts.

When this patch has completed successfully, the operating system on all management nodes is up-to-date.

Patching the Management Node Operating System

> **Caution:**
>
> Follow this procedure only when the appliance is running software version 3.0.2-b892153 or later. Otherwise, additional steps are required to move the primary role in the cluster between node patch operations. Follow this procedure instead: Patching the Management Node Operating System with Appliance Software 3.0.2-b852928 or Earlier

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

The Oracle Linux host operating system of the management nodes must be patched one node at a time; a rolling patch of all management nodes is not possible. This patching process, which involves updating the kernel and system packages, detects which of the nodes in the three-management-node cluster owns the virtual IP and the primary role. When the current primary node is patched, the primary role is first transferred to another node in the cluster. This configuration change occurs in the background and requires no separate or additional intervention from an administrator.

> **Note:**
>
> In case the ILOM also needs to be patched, you can integrate it into this procedure by executing the optional steps. The combined procedure eliminates the need to evacuate and reboot the same node twice.

You must patch management nodes one at a time, using each one's internal IP address as a command parameter. To obtain the host IP addresses, use one of these Service CLI commands:

- Enter `show ManagementNode name=<node_name>` and look for the `Ip Address` in the output.

```
PCA-ADMIN> show ManagementNode name=pcamn01
[...]
Data:
  Id = d9f10197-9a7a-4602-8440-f5f43f573f65
  Type = ManagementNode
  HW Id = AK0MYPCA3X
  MAC Address = a8:69:8c:05:e0:d8
  Ip Address = 100.96.2.33
  ILOM Ip Address = 100.96.0.33
  ILOM MAC Address = A8:69:8C:05:E0:DB
  [...]
```

- On systems running software version 3.0.2-b892153 or later, enter `getServerIP hostName=<node_name>`.

```
PCA-ADMIN> getServerIP hostName=pcamn01
[...]
Data:
  status = success
  data = 100.96.2.33
```

You cannot complete all of the patching tasks required in the Service Web UI for this component. Use the Service CLI to patch the management nodes.

> **Note:**
>
> In software version 3.0.2-b892153 or later all patch operations are based on the upgrade plan, which is generated when the pre-upgrade command is executed. For more information, see Prepare for Patching. When a component is already at the required version, the patch operation is skipped. However, patching with the same version can be forced using the Service Web UI or Service CLI command option (`force=True`), if necessary.

Using the Service CLI

1. Get the IP address of the management node for which you intend to upgrade the host operating system.
2. Optionally, patch the server ILOM first.
   a. Enter the ILOM patch command.
   Syntax (entered on a single line):

   ```
   patchIlom
   hostIp=<ilom-ip>
   ```

   Example:

   ```
   PCA-ADMIN> upgradeIlom hostIp=100.96.0.33
   Data:
     Service request has been submitted. Upgrade Job Id = 1632990827394-
   ilom-21089 Upgrade Request Id = UWS-1a97a8d9-9f06-a0c0-b972-d093bee40010

   PCA-ADMIN> getUpgradeJob upgradeJobId=1632990827394-ilom-21089
   ```

   b. Wait 5 minutes to allow the ILOM patch job to complete. Then proceed to patching the host.
3. Enter the management node host patch command.

   Syntax (entered on a single line):

   ```
   patchHost hostIp=<management-node-ip>
   ```

   Example:

   ```
   PCA-ADMIN> patchHost hostIp=100.96.2.33
   Command: patchHost hostIp=100.96.2.33
   Status: Success
   Time: 2022-01-01 21:06:56.849 UTC
   Data: Service request has been submitted.  Upgrade Job ID = 1632990827394-
   host-56156 \
   Upgrade Request ID = UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
   ```

4. Use the job ID to check the status of the patch process. The job ID is listed in the output of the patch command.

   ```
   PCA-ADMIN> getUpgradeJob upgradeJobId=1632990827394-host-56156
   ```

   At the end of the patching process, the management node is rebooted automatically. Wait approximately 5 minutes until the management node restarts.
5. When the first management node host operating system patch has completed successfully, execute the same command for the second management node.

When that patch has completed successfully, execute the same command for the third management node.

```
PCA-ADMIN> patchHost hostIp=100.96.2.35

PCA-ADMIN> patchHost hostIp=100.96.2.34
```

When all three operating system patch operations have completed successfully, the management node cluster is up-to-date.

> **Note:**
>
> After patching, if the upgrade plan specifies that the management nodes must be rebooted for the changes to take effect, a reboot is performed as part of the patch process. No administrator action is required.
>
> If the appliance is running software version 3.0.2-b892153 or earlier, all management nodes are rebooted as part of the patch process.

Patching the MySQL Cluster Database

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

The MySQL Cluster database is patched independently of the management node host operating system; the MySQL packages are deliberately kept separate from the Oracle Linux upgrade.

> **Note:**
>
> In software version 3.0.2-b892153 and later all patch operations are based on the upgrade plan, which is generated when the pre-upgrade command is executed. For more information, see Prepare for Patching. When a component is already at the required version, the patch operation is skipped. However, patching with the same version can be forced using the Service Web UI or Service CLI command option (`force=True`), if necessary.

Ensure you perform a system backup before you apply a patch. See the Backup and Restore section of the Oracle Private Cloud Appliance Administrator Guide.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.

3. Select the appropriate patch request type: Patch MySQL.
4. If required, fill out the patch request parameters:
   - **ULN:** Enter the fully qualified domain name of the ULN mirror in your data center. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **Advanced Options JSON:** Not available.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the patch command.

```
PCA-ADMIN> patchMySQL
Command: patchMySQL
Status: Success
Time: 2022-01-24 18:46:56.849 UTC
Data: Service request has been submitted.  Upgrade Job ID = 1642593347925-
mysql-40566 \
Upgrade Request ID = UWS-1ee38895-dedf-41c5-ab77-eebe294707ed
```

2. Use the request ID and the job ID to check the status of the patch process.

```
PCA-ADMIN> getupgradejobs
Command: getupgradejobs
Status: Success
Time: 2023-01-24 18:53:22,117 UTC
Data:
  id
upgradeRequestId                            commandName   result
  --
----------------                            -----------   ------
  1642593347925-mysql-40566        UWS-1ee38895-dedf-41c5-ab77-
eebe294707ed   mysql         Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1642593347925-mysql-40566
Command: getUpgradeJob upgradeJobId=1642593347925-mysql-40566
Status: Success
Time: 2023-01-24 18:54:05,408 UTC
Data:
  Upgrade Request Id = UWS-1ee38895-dedf-41c5-ab77-eebe294707ed
  Name = mysql
[...]
```

Patching Etcd and Vault

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

The secret service contains two components that need to be patched separately in this particular order: first Etcd, then Vault.

The Etcd and Vault patches are rolling patches: each patch is executed on all three management nodes with one command.

> **Note:**
>
> In software version 3.0.2-b892153 and later all patch operations are based on the upgrade plan, which is generated when the pre-upgrade command is executed. For more information, see Prepare for Patching. When a component is already at the required version, the patch operation is skipped. However, patching with the same version can be forced using the Service Web UI or Service CLI command option (`force=True`), if necessary.

Ensure you perform a system backup before you apply a patch. See the Backup and Restore section of the Oracle Private Cloud Appliance Administrator Guide.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.
3. Select the appropriate patch request type: Patch Etcd.
4. If required, fill out the patch parameters:
   - **ULN:** Enter the fully qualified domain name of the ULN mirror in your data center. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **Advanced Options JSON:** Not available.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.
6. When the Etcd patch has completed successfully, repeat this procedure to create a patch for Vault.

Using the Service CLI

1. Enter the two patch commands. Wait until the Etcd patch operation is finished before starting the Vault patch operation.

```
PCA-ADMIN> patchEtcd
Command: patchEtcd
Status: Success
Time: 2022-01-24 18:43:56.849 UTC
Data: Service request has been submitted.  Upgrade Job ID = 1642593966208-
etcd-6066 \
Upgrade Request ID = UWS-1ee38895-dedf-41c5-ab77-eebe294707ed

PCA-ADMIN> patchVault
Command: patchVault
```

```
Status: Success
Time: 2022-01-24 18:48:21.841 UTC
Data: Service request has been submitted.  Upgrade Job ID = 1642594274785-
vault-29202 \
Upgrade Request ID = UWS-77bc0c30-7ff5-4c50-ad09-6f96907e22e1
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getupgradejobs
Command: getupgradejobs
Status: Success
Time: 2023-01-24 21:23:22,117 UTC
Data:
  id
upgradeRequestId                          commandName   result
  --
----------------                          -----------   ------
  1642594274785-vault-29202        UWS-77bc0c30-7ff5-4c50-
ad09-6f96907e22e1   vault         Passed
  1642593966208-etcd-6066          UWS-1ee38895-dedf-41c5-ab77-
eebe294707ed   etcd          Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1642594274785-vault-29202
Command: getUpgradeJob upgradeJobId=1642594274785-vault-29202
Status: Success
Time: 2023-01-24 21:55:43,804 UTC
Data:
  Upgrade Request Id = UWS-77bc0c30-7ff5-4c50-ad09-6f96907e22e1
  Name = vault
[...]
```

Patching the Kubernetes Cluster

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

The Kubernetes container orchestration environment patching is also kept separate from the operating system. With a single command, all Kubernetes packages, such as kubeadm, kubectl and kubelet, are patched on the three management nodes and all the compute nodes. Note that this patching does not include the microservices running in Kubernetes containers.

> **Note:**
>
> In software version 3.0.2-b892153 or later all patch operations are based on the upgrade plan, which is generated when the pre-upgrade command is executed. For more information, see Prepare for Patching. When a component is already at the required version, the patch operation is skipped. However, patching with the same version can be forced using the Service Web UI or Service CLI command option (`force=True`), if necessary.

Ensure synchronization of the mirror on the shared storage is complete prior to Kubernetes patching by issuing the `syncUpstreamUlnMirror` command. For more information, see Prepare for Patching.

About the Kubernetes Upgrade Process

To ensure compatibility and continuation of service, Kubernetes must be upgraded one version at a time. Skipping versions – major or minor – is not supported. The Private Cloud Appliance Upgrader manages this process by upgrading or patching all parts of the Kubernetes cluster to the next available version, repeating the same sequence of operations until the entire environment runs the latest Kubernetes version available from the appliance software repositories.

Upgrading or patching the Kubernetes cluster is a time-consuming process that involves the Private Cloud Appliancemanagement nodes and compute nodes. Each additional compute node extends the process by appoximately 10 minutes for each incremental version of Kubernetes.

With appliance software version 3.0.2-b925538, the container orchestration environment is upgraded or patched from Kubernetes version 1.20.x to version 1.25.y, meaning the entire process must run 5 times. After each successful run, the repository is synchronized to retrieve the next required version. However, with this version of the appliance software the repository is reconfigured to allow multiple versions of the Kubernetes packages, so the resync will no longer be required.

Each individual Kubernetes node upgrade is expected to take around 10 minutes. Testing indicates that upgrading or patching the Private Cloud Appliance Kubernetes cluster from version 1.20 to version 1.25 takes approximately 4-5 hours for a base rack configuration with 3 management nodes and 3 compute nodes. On a full rack with 20 compute nodes the entire process requires at least 9 hours and may take up to 18 hours to complete. The estimated time for the rack's specific configuration is reported in the upgrade plan.

To monitor the upgrade or patching progress, periodically check the job status or the logs.

- Check job status through the Service CLI: `getUpgradeJob upgradeJobId=`***`<id>`***
- View Upgrader logs on a management node: `tail -f /nfs/shared_storage/pca_upgrader/log/pca-upgrader_kubernetes_cluster_`***`<time_stamp>`***`.log`.

During Kubernetes upgrade or patching, certain services could be temporarily unavailable.

- The Compute Web UI, Service Web UI, OCI CLI, and Service CLI can all become temporarily unavailable. Users should wait a few minutes before attempting their operations again. Administrative operations in the Service Enclave (UI or CLI) must be avoided during upgrade or patching.
- When the Kubernetes upgrade is initiated, the Kubernetes Workload Monitoring Operator (Sauron service) is taken down. As a result, the Grafana, Prometheus, and other Sauron ingress endpoints cannot be accessed. They become available again after both the Kubernetes cluster and the containerized microservices (platform layer) upgrade or patching processes have been completed.

Managing Unprovisioned Compute Nodes

If you upgrade or patch the Kubernetes cluster on a Private Cloud Appliance that contains unprovisioned compute nodes, there could be provisioning issues later. Because those compute nodes were not part of the Kubernetes cluster when the newer version was applied, you may need to rediscover them first.

If compute node provisioning fails after upgrading or patching the Kubernetes cluster, log on to one of the management nodes using ssh. Rediscover the unprovisioned compute nodes by running the following command with the appropriate host names:

```
# pca-admin compute node rediscover --hostname pcacn000
```

When the compute nodes have been rediscovered, provisioning is expected to work as intended.

For more information about provisioning, refer to "Performing Compute Node Operations" in the chapter Hardware Administration of the Oracle Private Cloud Appliance Administrator Guide.

Patch the Kubernetes Cluster Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.
3. Select the appropriate patch request type: Patch Kubernetes.
4. If required, fill out the patch parameters:
   - **ULN:** Enter the fully qualified domain name of the ULN mirror in your data center. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **Advanced Options JSON:** Not available.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.

Patch the Kubernetes Cluster Using the Service CLI

1. Enter the patch command.

```
PCA-ADMIN> patchKubernetes
Command: patchKubernetes
Status: Success
Time: 2022-01-18 20:02:05,408 UTC
Data: Service request has been submitted.  Upgrade Job ID = 1642509549088-
kubernetes-51898 \
Upgrade Request ID = UWS-4f0d9e99-a515-4170-ab35-9f8bdcbdb2b5
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getupgradejobs
Command: getupgradejobs
Status: Success
Time: 2023-01-22 19:52:16,398 UTC
Data:
  id
upgradeRequestId                            commandName   result
  --
----------------                            -----------   ------
  1642509549088-kubernetes-51898   UWS-4f0d9e99-a515-4170-
ab35-9f8bdcbdb2b5   kubernetes    Passed
  1642492793827-oci-12162          UWS-6e06bbb7-16b8-49ba-9c33-
```

```
f42fffbe1323   oci           Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1642509549088-kubernetes-51898
Command: getUpgradeJob upgradeJobId=1642509549088-kubernetes-51898
Status: Success
Time: 2023-01-22 20:11:43,804 UTC
Data:
  Upgrade Request Id = UWS-4f0d9e99-a515-4170-ab35-9f8bdcbdb2b5
  Name = kubernetes
[...]
```

Patching the Platform

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

The platform patching covers both the internal services of the platform layer, and the administrative and user-level services exposed through the infrastructure services layer.

The containerized microservices have their own separate patching mechanism. A service is patched if a new Helm deployment chart and container image are found in the `pca302_containers` ULN channel. When a new deployment chart is detected during the patching process, the pods running the services are restarted with the new container image.

> **Note:**
>
> In software version 3.0.2-b892153 and later all patch operations are based on the upgrade plan, which is generated when the pre-upgrade command is executed. For more information, see Prepare for Patching. When a component is already at the required version, the patch operation is skipped. However, patching with the same version can be forced using the Service Web UI or Service CLI command option (`force=True`), if necessary.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.
3. Select the appropriate patch request type: Patch Platform.
4. If required, fill out the patch parameters:
   - **ULN:** Enter the fully qualified domain name of the ULN mirror in your data center. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **Advanced Options JSON:** Not available.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".

5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the patch command.

```
PCA-ADMIN> patchPlatform
Command: patchPlatform
Status: Success
Time: 2022-12-08 17:26:12,217 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1638984971208-
platform-79257 \
Upgrade Request Id = UWS-39f3f08f-b2d1-4804-8185-2dd3af60dd41
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getupgradejobs
Command: getupgradejobs
Status: Success
Time: 2022-12-08 17:36:34,657 UTC
Data:
  id
upgradeRequestId                          commandName   result
  --
----------------                          -----------   ------
  1638984971208-platform-79257     UWS-39f3f08f-
b2d1-4804-8185-2dd3af60dd41   platform      None

PCA-ADMIN> getupgradejob upgradeJobId=1638984971208-platform-79257
Command: getupgradejob upgradeJobId=1638984971208-platform-79257
Status: Success
Time: 2022-12-08 17:38:19,385 UTC
Data:
  Upgrade Request Id = UWS-39f3f08f-b2d1-4804-8185-2dd3af60dd41
  Name = platform
  Start Time = 2022-12-08T17:26:11
  Pid = 79257
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_platform_services_2022_12_08-17.26.11.log
  Arguments =
{"component_names":null,"diagnostics":false,"display_task_plan":false,"dry_ru
n_tasks":false, \

"expected_iso_checksum":null,"fail_halt":false,"fail_upgrade":null,"image_loc
ation":null, \
[...]
  Process = alive
  Tasks 1 - Name = Validate ULN Channel URL
  Tasks 1 - Description = Verify that the ULN channel URL is accessible
  Tasks 1 - Time = 2022-12-08T17:26:12
[...]
```

Patching Firmware

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

Firmware is included in the ISO image for all component ILOMs, for the Oracle ZFS Storage Appliance, and for the switches. Select the instructions below for the component type you want to patch.

> **Note:**
>
> In software version 3.0.2-b892153 and later all patch operations are based on the upgrade plan, which is generated when the pre-upgrade command is executed. For more information, see Prepare for Patching. When a component is already at the required version, the patch operation is skipped. However, patching with the same version can be forced using the Service Web UI or Service CLI command option (`force=True`), if necessary.

Obtaining an ILOM IP Address

Using the Service Web UI

1. In the navigation menu, click Rack Units.
2. Click on the name of the component you are patching.
3. Select the Rack Unit Information tab.
4. Record the IP Address listed under ILOM IPs.

Using the Service CLI

1. To list the ILOM IP addresses of all management nodes or compute nodes, use these commands:

```
PCA-ADMIN> getCNIloms
Data:
  status = success
  data 1 = 100.96.0.66
  data 2 = 100.96.0.64
  data 3 = 100.96.0.65

PCA-ADMIN> getMNIloms
Data:
  status = success
  data 1 = 100.96.0.33
  data 2 = 100.96.0.34
  data 3 = 100.96.0.35
```

2. To obtain the ILOM IP address that corresponds with a specific node host name, use the following command:

```
PCA-ADMIN> getServerILOMIP hostName=<node_name>
```

For example:

```
PCA-ADMIN> getServerILOMIP hostName=pcacn002
Data:
  status = success
  data = 100.96.0.65

PCA-ADMIN> getServerILOMIP hostName=pcamn03
Data:
  status = success
  data = 100.96.0.35
```

Using the Service CLI with Appliance Software 3.0.2-b852928 or Earlier

1. Find the component ID:

Syntax (entered on a single line):

```
list <component>
```

Example:

```
PCA-ADMIN> list computeNode
Command: list computeNode
Status: Success
Time: 2021-12-17 21:30:41,064 UTC
Data:
  id                                   name       provisioningState
provisioningType
  --                                   ----       -----------------
----------------
  03111396-bb33-4249-9561-b921387c6f3a   pcacn003   Provisioned         KVM
  1600443b-00f3-4424-946d-bd52df778aaf   pcacn001   Provisioned         KVM
  69e4e3b7-9390-4283-b246-49ebedccac95   pcacn002   Provisioned         KVM
```

2. Use the component ID to show the details of that component.

```
PCA-ADMIN> show computeNode id=03111396-bb33-4249-9561-b921387c6f3a
Command: show computeNode id=03111396-bb33-4249-9561-b921387c6f3a
Status: Success
Time: 2021-12-17 21:42:47,724 UTC
Data:
  Id = 03111396-bb33-4249-9561-b921387c6f3a
  Type = ComputeNode
  Provisioning State = Provisioned
[...]
  Ip Address = 100.96.2.64
  ILOM Ip Address = 100.96.0.64
  Hostname = pcacn001
[...]
```

Patching ILOMs

> **Note:**
>
> In case a server node needs both the host operating system and the ILOM to be patched, you can avoid having to reboot the same node twice by combining the two patch operations. Instructions are provided in the following sections:
>
> - Patching a Compute Node
> - Patching the Management Node Operating System

ILOM patches can be applied to management nodes and compute nodes; the firmware packages might be different per component type. You must patch ILOMs one at a time, using each one's internal IP address as a command parameter.

> **Caution:**
>
> You must NOT patch the ILOM of the management node that holds the management virtual IP address, and thus the primary role in the cluster. To determine which management node has the primary role in the cluster, and make another node the primary, use the following Service CLI commands:
>
> ```
> PCA-ADMIN> getPrimaryMgmtNode
>   status = success
>   data = pcamn01
>
> PCA-ADMIN> updatePrimaryNode node=pcamn02
> Data:
>   status = success
>   message = Successfully issued update primary node command
>
> PCA-ADMIN> getPrimaryMgmtNode
>   status = success
>   data = pcamn02
> ```

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.
3. Select the appropriate patch request type: Patch ILOM.
4. Fill out the patch parameters:
   - **ULN:** This parameter is deprecated.
   - **Host IP:** Enter the component's assigned IP address in the ILOM network.
   - **Advanced Options JSON:** Not available.

- **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".

5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.

Using the Service CLI

1. Get the IP address of the ILOM for which you intend to patch the firmware.
2. Enter the patch command.

   Syntax (entered on a single line):

   ```
   patchIlom
   hostIp=<ilom-ip>
   ```

   Example:

   ```
   PCA-ADMIN> patchIlom hostIp=100.96.0.62
   Command: patchIlom hostIp=100.96.0.62
   Status: Success
   Time: 2022-01-24 18:18:31,044 UTC
   Data:
     Service request has been submitted. Upgrade Job Id = 1620921089806-
   ilom-21480 Upgrade Request Id = UWS-732d6fce-9f06-4329-b972-d093bee40010
   ```

3. Use the request ID and the job ID to check the status of the upgrade process.

   ```
   PCA-ADMIN> getUpgradeJobs
     id
   upgradeRequestId                          commandName   result
     --
   ----------------                          -----------   ------
     1620921089806-ilom-21480         UWS-732d6fce-9f06-4329-b972-
   d093bee40010   ilom          Passed

   PCA-ADMIN> getUpgradeJob upgradeJobId=1620921089806-ilom-21480
   Command: getUpgradeJob upgradeJobId=1620921089806-ilom-21480
   Status: Success
   Time: 2023-01-18 18:23:39,690 UTC
   Data:
     Upgrade Request Id = UWS-732d6fce-9f06-4329-b972-d093bee40010
     Name = ilom
   [...]
   ```

4. Use the `syncHardwareData` command to update the hardware attributes in the system hardware database.

> **Note:**
>
> The `syncHardwareData` command is also used for internal automated system tasks. If this automated task is running when you issue the `syncHardwareData` command manually, a lock will prevent your command from running and you could see this error:
>
> ```
> This command cannot be performed at this time. Please try again.
> ```
>
> Wait a few moments, then re-issue the `syncHardwareData` command.

At the end of the patch, the ILOM itself is rebooted automatically. However, the server component also needs to be rebooted for all changes to take effect. Wait 5 minutes to allow the ILOM workflow to complete first.

For minimum operational impact, schedule the compute node and management node reboot operations after all ILOMs have been patched. Take into account that rebooting the compute nodes requires migrating the compute instances. For more information, refer to "Performing Compute Node Operations" in the Hardware Administration chapter of the Oracle Private Cloud Appliance Administrator Guide.

Patching the ZFS Storage Appliance Operating Software

To patch the operating software of the system's ZFS Storage Appliance, you only need to provide the path to the ULN mirror. The IP addresses of the storage controllers are known, and a single command initiates a rolling patch of both controllers.

> **Caution:**
>
> Ensure that no users are not logged in to the ZFS Storage Appliance or the storage controller ILOMs during the patching process.
>
> Do not make storage configuration changes while an upgrade is in progress. While controllers are running different software versions, configuration changes made to one controller are not propagated to its peer controller.
>
> During firmware patching the storage controllers are placed in active/passive mode. They automatically return to active/active after patching is completed.

> **Note:**
>
> ZFS Storage Appliance updates may include ILOM and BIOS firmware. If an update to the BIOS firmware is required, there will be a note in the Upgrader log indicating that the BIOS will be updated the next time the storage controller is rebooted.

Before You Begin

Before you start patching a ZFS Storage Appliance, you must disable the node state service to prevent errors in node states after the patch operation.

1. From a management node, set the provisioning lock by issuing this command:

   ```
   pca-admin locks set system provisioning
   ```

2. Perform the ZFS Storage Appliance patch operation using either the Service Web UI or the Service CLI procedure below.
3. Release the provisioning lock.

   ```
   pca-admin locks unset system provisioning
   ```

4. Confirm the lock state.

   ```
   pca-admin locks show system
   ```

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.
3. Select the appropriate patch request type: Patch Zfssa.
4. Fill out the patch parameters:
   - **ULN:** This parameter is deprecated.
   - **Advanced Options JSON:** Not available.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the patch command.

```
PCA-ADMIN> patchZfssa
Command: patchZfssa
Status: Success
Time: 2022-01-24 18:15:07,453 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1643035466051-
zfssa-62915 Upgrade Request Id = UWS-831fd008-cc32-428d-8e76-91c43081f6e7
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
Status: Success
Time: 2023-01-24 18:19:29,731 UTC
Data:
  id
upgradeRequestId                            commandName   result
  --
----------------                            -----------   ------
  1643035466051-zfssa-62915        UWS-831fd008-
cc32-428d-8e76-91c43081f6e7   zfssa         Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1643035466051-zfssa-62915
Command: getUpgradeJob upgradeJobId=1643035466051-zfssa-62915
Status: Success
Time: 2022-01-24 18:21:52,775 UTC
Data:
  Upgrade Request Id = UWS-831fd008-cc32-428d-8e76-91c43081f6e7
  Name = zfssa
[...]
```

Patching the Switch Software

The appliance rack contains three categories of Cisco Nexus switches: a management switch, two leaf switches, and two spine switches. They all run the same Cisco NX-OS network operating software. **You must apply the patches in this order: leaf switches first, then spine switches, and finally the management switch**. Only one

command per switch category is required, meaning that the leaf switches and the spine switches are patched in pairs.

Some versions of the network operating software consist of two files: a binary file and an additional EPLD (electronic programmable logic device) image. Both are automatically retrieved from their designated location during the patch process, and applied in the correct order.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Patch* as the Request Type.
3. Select the appropriate patch request type: Patch Switch.
4. Fill out the patch parameters:
   - **ULN:** This parameter is deprecated.
   - **Advanced Options JSON:** Not available.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
   - **Switch Type:** Select the switch type you intend to patch. The preferred order is as follows: leaf switches first, then spine switches, and finally the management switch.
5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.
6. When the patch has completed successfully, but other switches in the system still need to be patched, repeat this procedure for any other type of switch that requires patching.

Using the Service CLI

1. Determine the type of switch to patch (spine, leaf, management).
2. Enter the patch command.

   Syntax (entered on a single line):

   ```
   patchSwitch
   switchType=[MGMT, SPINE, LEAF]
   ```

   Example:

   ```
   PCA-ADMIN> patchSwitch switchType=LEAF
   Command: patchSwitch switchType=LEAF
   Status: Success
   Time: 2023-01-24 18:16:54,704 UTC
   Data:
     Service request has been submitted. Upgrade Job Id = 1630511206512-cisco-20299
   Upgrade Request Id = UWS-44688fe5-b4f8-407f-a1b5-8cd1b685c2c3
   ```
3. Use the request ID and the job ID to check the status of the upgrade process.

   ```
   PCA-ADMIN> getUpgradeJobs
   Status: Success
   Time: 2023-01-24 18:19:29,731 UTC
   Data:
     id                                  upgradeRequestId
   commandName   result
   ```

```
  --
----------------                                  -----------   ------
  1632914107346-zfssa-83002
UWS-881af57f-5dfb-4c75-8026-9f00cf3eb7c9   zfssa         Passed
  1630511206512-cisco-20299        UWS-44688fe5-b4f8-407f-
a1b5-8cd1b685c2c3   cisco         Passed
  1620921089806-ilom-21480         UWS-732d6fce-9f06-4329-b972-
d093bee40010   ilom          Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1630511206512-cisco-20299
Command: getUpgradeJob upgradeJobId=1630511206512-cisco-20299
Status: Success
Time: 2023-01-24 18:27:52,083 UTC
Data:
  Upgrade Request Id = UWS-44688fe5-b4f8-407f-a1b5-8cd1b685c2c3
  Name = cisco
[...]
```

4. Use the `syncHardwareData` command to update the hardware attributes in the system hardware database.

> **Note:**
>
> The `syncHardwareData` command is also used for internal automated system tasks. If this automated task is running when you issue the `syncHardwareData` command manually, a lock will prevent your command from running and you could see this error:
>
> ```
> This command cannot be performed at this time. Please try again.
> ```
>
> Wait a few moments, then re-issue the `syncHardwareData` command.

Patching Oracle Cloud Infrastructure Images

> **Caution:**
>
> Ensure that all preparation steps for system patching have been completed. For instructions, see Prepare for Patching.

When new Oracle Cloud Infrastructure Images become available and supported for Oracle Private Cloud Appliance, you can pick up these images using the patching process.

Oracle Cloud Infrastructure Images installed using the patching method are stored in the `/nfs/shared_storage/oci_compute_images` directory on the ZFS storage appliance.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

The Create Request window appears. Choose *Patch* as the Request Type.

3. Select the appropriate patch request type: Patch OCIImages.
4. If required, fill out the request parameters:
   - **ULN:** Enter the fully qualified domain name of the ULN mirror in your data center. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **Advanced Options JSON:** Not available.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new patch request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the patch command.

```
PCA-ADMIN> patchOCIImages
Command: patchOCIImages
Status: Success
Time: 2023-01-18 19:33:09,756 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1641839285475-oci-94665 \
Upgrade Request Id = UWS-778b08bc-f579-492b-993d-915dcf581374
```

2. Use the request ID and the job ID to check the status of the patching process.

```
PCA-ADMIN> getupgradejobs
Command: getupgradejobs
Status: Success
Time: 2023-01-18 22:38:51,764 UTC
Data:
  id                              upgradeRequestId
commandName   result
  --                              ----------------
-----------   ------
  1641839285475-oci-94665         UWS-778b08bc-f579-492b-993d-915dcf581374
oci           Passed
  1641838937541-platform-56313    UWS-bc4372ae-8f51-4b40-9306-992fb6459878
platform      Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1680260388058-oci-94665
Command: getUpgradeJob upgradeJobId=1680260388058-oci-94665
Status: Success
Time: 2023-01-18 23:03:22,769 UTC
Data:
  Upgrade Request Id = UWS-778b08bc-f579-492b-993d-915dcf581374
  Name = oci
[...]
```