

BigFix
Patches for Oracle Linux - User's Guide



Special notice

Before using this information and the product it supports, read the information in [Notices \(on page xlix\)](#).

Edition notice

This edition applies to BigFix version 10 and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Special notice.....	ii
Edition notice.....	iii
Chapter 1. Overview.....	6
Supported architecture, erratas, and repositories.....	6
Site applicability matrix.....	9
Fixlet fields.....	9
Chapter 2. Setup.....	11
Site subscription.....	11
Subscribing to the BigFix Patches for Oracle Linux site.....	11
Setting up a local repository.....	11
Chapter 3. Using Patches for Oracle Linux.....	12
Patch using Fixlets.....	12
Patch by using the YUM utility.....	12
Supersedence.....	12
Chapter 4. Multiple-Package Baseline Installation.....	13
Installing multiple packages in a baseline.....	14
Chapter 5. Manage YUM transactions.....	16
Rolling back a YUM transaction.....	18
Undoing a YUM transaction	18
Redo a YUM transaction.....	19
Checking for YUM package updates.....	20
Chapter 6. Manage custom repositories.....	21
Registering a repository.....	21
Unregistering a repository from an endpoint.....	22
Adding repositories.....	22
Importing a repository.....	22
Chapter 7. Using the download plug-in.....	24
Manage Download Plug-ins dashboard overview.....	25
Registering the Oracle linux Download Plug-in.....	26
Configuring the basic Oracle linux Download Plug-in settings.....	28
Configuring the advanced Oracle linux Download Plug-in settings.....	30

Unregistering the Oracle linux Download Plug-in.....	32
Upgrading the Oracle linux Download Plug-in	33
Extending the Oracle linux Download Plug-in.....	33
Chapter 8. Using the download cacher.....	37
Oracle linux Download Cacher usage information.....	37
Using the Oracle linux Download Cacher for air-gapped environments.....	42
Caching packages on the sha1 folder.....	43
Caching packages on the local cache folder.....	44
Chapter 9. Frequently asked questions.....	45
Appendix A. Support.....	48
Notices.....	xlix

Chapter 1. Overview

BigFix® Patches for Oracle Linux keeps your Linux™ clients current with the latest updates and service packs.

Patch management is available through the Patches for Oracle Linux sites from BigFix. For each new patch or update that becomes available, BigFix releases a Fixlet that can identify and remediate all the computers in your enterprise that need it. With a few keystrokes, the BigFix Console Operator can apply the patch to all the relevant computers and visualize its progress as it deploys throughout the network.

The BigFix agent checks the operating system version, file versions, the language of the system and other relevant factors to determine when and if a patch is necessary.

Fixlets allow you to manage large numbers of updates and patches with comparative ease, enabling automated, highly targeted deployment on any schedule that you want. Large downloads can be phased to optimize network bandwidth and the entire deployment process can be monitored, graphed, and recorded for inventory or audit control. Fixlets often have extra notes that allow the Console Operator to work around issues. Once you have subscribed to the Patches for Oracle Linux sites, you can do the following:

- Patch using Fixlets
- Identify available YUM packages through a task.
- Roll back, undo, and redo transactions in your deployment through the YUM Transaction History dashboard.
- Register, add, unregister, or import custom repositories through the Oracle Linux Custom Repository Management dashboard.



Note: Local repositories must be set up separately.

New Features

Users of Oracle Linux 9 can now use the feature of the download plug-in and the download cacher along with the Custom Repository Management dashboard and YUM Transaction History dashboard.



Note: Download plug-in and the download cacher are not applicable for Oracle Linux 6, Oracle Linux 7, and Oracle Linux 8.

Supported architecture, erratas, and repositories

The various features of Patches for Oracle Linux apply differently for each Oracle Linux site.

BigFix Patches for Oracle Linux supports security, bug fix, and enhancement erratas of Oracle Linux. An errata is an individual package update that Oracle releases for important changes to Oracle Linux. Errata packages can contain security, bug fix, and feature enhancement advisories. To learn more about Oracle Linux erratas, see https://docs.oracle.com/cd/E37670_01/E37355/html/ch03s03.html. BigFix Patches for Oracle Linux supports erratas that are released for the following Oracle Linux repositories.

Table 1. Applicable features for each Patches for Oracle Linux site

Patches for Oracle Linux site	Supported architecture	Supported repositories
Patches for Oracle Linux 6	X86-64, i386	<ul style="list-style-type: none"> • Latest • UEK Release 4 • UEK Release 3 • UEK Release 2 • Add-ons • OFED (UEK Release 4) • OFED 2.0 • OpenStack 1.0 • Ceph 1.0 • Spacewalk 2.4 Server • Spacewalk 2.2 Server • Spacewalk 2.0 Server • Spacewalk 2.4 Client • Spacewalk 2.2 Client • Spacewalk 2.0 Client • Software Collection 1.2 • MySQL 5.7 • MySQL 5.6 • MySQL 5.5 • GDM Multiseat
Patches for Oracle Linux 7	X86-64	<ul style="list-style-type: none"> • Latest • UEK Release 6 • UEK Release 5 • UEK Release 4 • UEK Release 3 • Optional Latest • OFED (UEK Release 4) • OFED 2.0 • Add-ons • OpenStack 2.0 • Ceph 1.0 • Spacewalk 2.2 Client • Software Collection 1.2 • MySQL 5.7

Table 1. Applicable features for each Patches for Oracle Linux site (continued)

Patches for Oracle Linux site	Supported architecture	Supported repositories
		<ul style="list-style-type: none"> • MySQL 5.6 • MySQL 5.5
Patches for Oracle Linux 8	x86-64	<ul style="list-style-type: none"> • Latest • Appstream • Addons • Codeready-builder • BaseOS GA • 8.1 BaseOS • 8.2 BaseOS • Spacewalk Client 2.10 • UEK Release 6
Patches for Oracle Linux 9	x86-64	<ul style="list-style-type: none"> • Latest • Appstream • UEKR7 • Addons

To browse the list of available Oracle Linux packages, see the following references.

- Oracle Linux 6 package repositories: see <http://yum.oracle.com/oracle-linux-6.html>.
- Oracle Linux 7 package repositories, see <http://public-yum.oracle.com/oracle-linux-7.html>.
- Oracle Linux 8 package repositories, see <https://public-yum.oracle.com/oracle-linux-8.html>
- Oracle Linux 9 package repositories, see <https://public-yum.oracle.com/oracle-linux-9.html>



Note: You must use BigFix version 9.5 and later to use BigFix Patch for Oracle Linux.

BigFix Patch uses the BigFix RHEL agent version 9.5.2.

BigFix supports both the Red Hat Compatible kernel and Unbreakable Enterprise Kernel (UEK) that Oracle Linux offers.



Note: Some repositories are not enabled by default since they might break Oracle Enterprise Linux upstream compatibility. Users might encounter errors regarding missing packages. For more information, see [Frequently Asked Questions \(on page 45\)](#).

Site applicability matrix

The various features of Patches for Oracle Linux apply differently for each Oracle Linux site.

The following matrix shows the Patches for Oracle Linux features that apply for each Oracle Linux site.

Table 2. Applicable features for each Patches for Oracle Linux site

BigFix site	Custom Repository Management dashboard	YUM Transaction History dashboard
Patches for Oracle Linux 6	Applicable	Applicable
Patches for Oracle Linux 7	Applicable	Applicable
Patches for Oracle Linux 8	Applicable	Applicable
Patches for Oracle Linux 9	Applicable	Applicable

Fixlet fields

Fixlets contain fields of metadata that provide specific details.

Some Fixlet fields are common across all domains, that is, categories of BigFix sites. Other fields are common to only one domain or product, such as Patch Management.

The following table lists the Fixlet fields and their descriptions.

Table 3. Fixlet fields and descriptions

Fixlet fields	Description	BigFix domain
ID	A numerical ID assigned to the Fixlet by the author.	All
Name	The name assigned to the Fixlet by the author.	All
Applicable Computer Count	The number of BigFix clients in the network currently affected by the Fixlet.	All
Category	The type of Fixlet, such as a Security Patch or Update.	All
Download Size	The size of the remedial file or patch that the action downloads.	All
Source	The name of the source vendor that provides the Fixlet information.	All
Source ID	A numerical ID assigned to the Fixlet to relate it back to its source.	All
Source Release Date	The date when an upstream vendor releases the patch.	All

Table 3. Fixlet fields and descriptions (continued)

Fixlet fields	Description	BigFix domain
Source Severity	A measure of how critical a Fixlet is, assigned by the Fixlet author. Typical values are Critical, Important, Moderate, or Low.	All
Site	The name of the site that is generating the relevant Fixlet.	All
Unlocked Computer Count	The number of unlocked computers that are affected by the Fixlet.	All
Open Action Count	The number of distinct actions that are open for the given Fixlet.	All
X-Fixlet-product-family	The product family that the patch belongs to.	Windows Patching (Relates to BigFix Patch Management)
X-Fixlet-product	The product that the patch belongs to under a certain product family.	Windows Patching (Relates to BigFix Patch Management)
X-Fixlet-component	A component that the patch targets under a certain product family.	Windows Patching (Relates to BigFix Patch Management)
Modification Time	The time when a given Fixlet was last modified.	All
X-Fixlet-first-propagation	The Fixlet release date.	All

Chapter 2. Setup

Set up your environment for patch management.

Site subscription

Sites are collections of Fixlet messages that are created internally by you, by HCL, or by vendors.

Subscribe to a site to access the Fixlet messages to patch systems in your deployment.

You can add a site subscription by using the Licensing Dashboard. For more information about subscribing to Fixlet sites, see the *BigFix Installation Guide*.

For more information about sites, see the *BigFix Console Operator's Guide*.

Subscribing to the BigFix Patches for Oracle Linux site

Subscribe to the Patches for Oracle Linux sites through the License Overview dashboard.

1. From the **BigFix Management** domain, click **License Overview** dashboard.
2. Scroll down to the applicable Patches for Oracle Linux site and click **Enable**.
3. Open the **Manage Sites** node and select the applicable Patches for Oracle site. For example, Patches for Oracle Linux 7.
4. From the site dialog, click the **Computer Subscriptions** tab to assign the site to the appropriate computers.
5. From the **Operator Permissions** tab, select the operators that you want to associate with the site and their level of permission.
6. Click **Save Changes**.

You are now subscribed to a Patches for Oracle Linux site.

Setting up a local repository

Manage custom repositories through the Oracle Linux Custom Repository Management dashboard.

BigFix Patch for Oracle Linux supports the use of custom repositories. You can manage custom repositories through the Oracle Linux Custom Repository Management dashboard. However, the dashboard does not support the creation and maintenance of local repositories. You must create local repositories separately.

To learn more about creating local repositories, see <http://www.oracle.com/technetwork/articles/servers-storage-admin/yum-repo-setup-1659167.html>.

Chapter 3. Using Patches for Oracle Linux

You can manage Fixlets using dashboards, reports, and wizards. You can deploy, fix, and uninstall Fixlets. You can also view the breakdown of Fixlets available or needed in your deployment.

Patch using Fixlets

You can deploy patches from the BigFix Console. In the Patch Management domain in the console navigation tree, click OS Vendors and click Oracle Linux. Double-click the Fixlet that you want to deploy. Click the tabs at the top of the Fixlet window to review additional details, and then click the appropriate link in the Actions box to start deployment. Click **OK**.

Patch by using the YUM utility

Yellow dog Updater, Modified (YUM) is a package management tool that updates, installs, and removes Red Hat Package Manager (RPM) packages. YUM uses a command-line interface and simplifies the process of installing, uninstalling, and updating packages, provided that there is access to the YUM repository.

YUM utility configuration settings

The Patches for Oracle Linux site use YUM utility settings in the Fixlet settings in `/etc/yum.conf`. The following YUM configuration settings are not used in the Fixlet setting.

- cachedir
- keepcache
- plugins
- reposdir
- pluginpath
- pluginconfpath
- metadata_expire
- installonlypkgs

Supersedence

Please refer to Supersedence for Non-Windows to know more about the supersedence.

Chapter 4. Multiple-Package Baseline Installation

BigFix Patch provides a solution to combine the installation of updates for multiple packages in a baseline into a single task, which can reduce the execution time of the baseline.

Baselines can help you gather multiple Fixlets into groups, which you can apply immediately to any set of target computers. It is a powerful way to deploy a group of actions across an entire network. However, each Fixlet in a baseline creates a separate update transaction when the baseline is run. A single baseline can have numerous calls, which can severely impact performance as it increases the time taken to complete all the transactions.

The multiple-package baseline installation solution helps improve the performance that is due to the dependency resolution and package installation that is done separately for each Fixlet. This solution requires you to enable the feature at the start of the baseline and append the installation task to install the relevant packages from a single call.

Use the **Enable the Multiple-Package Baseline Installation feature** task to set the flag that instructs Fixlets to add packages to a list instead of installing them. The flag is cleared after the baseline is completed. You must add the appropriate task at the start of the baseline to allow the installation of multiple packages from a single command.

A multiple-package installation task is made available for each **OEL** distribution, operating system version, service pack level, and architecture. You must add the appropriate installation task at the end of your baseline to complete the dependency resolution, download the packages, and then install them on the endpoints.

Use the appropriate task to install the relevant packages listed on the `MultiPkgInstall.txt` file, located in the following directory locations from a single yum call.

On Windows systems:

```
%PROGRAM FILES%\BigFix Enterprise\BES Client\EDRDeployData
```

On Linux systems:

```
/var/opt/BESClient/EDRDeployData
```

The task action skips packages with broken dependencies.

These tasks must be run at the end of the baseline to do dependency resolution and package installation for the entire baseline in a single instance.



Note: Ensure that the following option is unchecked for the related tasks: **Baseline will be relevant on applicable computers where this component is relevant.**

You can also do a dry run of the installation to preview the changes on the packages to avoid broken dependencies, which might be due to undesired packages updates. The test action outputs to the following files at `/var/opt/BESClient/EDRDeployData`:

`PkgToInstallList.txt` **file**

This file contains packages that are to be installed after a dependency check.

Installing multiple packages in a baseline

The multiple-package baseline installation feature helps you to save time when deploying Fixlets with multiple unique packages from a baseline.

To install or update packages for all Fixlets in the baseline, you must add the task to enable the feature and add the appropriate multiple-package baseline installation task into the baseline.



Note: The multiple-package baseline installation feature does not support the pre-cache option **Start download before all constraints** are met in **Take Action**.

1. Create a baseline.

From the console, select **Create New Baseline** from the **Tools** menu. You can also highlight the Fixlets from a Fixlet site and select **Add to New Baseline** from the **context** menu.

2. Add the related tasks to the new baseline. The order of the tasks as listed in the tables is important.

OEL 9

Table 4. Tasks for OEL 9

Fixlet ID	Task
200	Delete OEL 9 Package List File for Multiple-Package Baseline Installation
300	TROUBLESHOOTING OEL 9 Patching Deployment Logs - Cleanup
301	Import RPM-GPG-KEY-oracle-release - Oracle Linux 9 x86_64
201	Enable the Multiple-Package Baseline Installation feature - Oracle Linux 9 x86_64

3. Selectively add the patch Fixlets in the baseline.

Ensure that for all Fixlets the **Baseline will be relevant on applicable computers where this component is relevant** option is selected.

4. Add the appropriate **Multiple-Package Baseline Installation** task at the end of the baseline. With this task, you can deploy any of the following actions:
 - Run a preview of the installation, without actually installing the packages, to check for possible issues.
 - Install all the RPMs in a single transaction.

Ensure that the **Baseline will be relevant on applicable computers where this component is relevant** option is not selected.



Note: When you deploy the baseline, the initial sub-action status for all the patch Fixlets will show that they failed. This is the expected behaviour. The process for downloading and installing the packages



in the baseline is not done at the Fixlet action level, but in the **Multiple-Package Baseline Installation** task. When the baseline completes, the baseline sub-action status of the Fixlets will reflect the final state of each patch installation.

Chapter 5. Manage YUM transactions

View YUM transaction history and manage transactions through the YUM Transaction History dashboard.

The dashboard displays the YUM transaction history and can be used to roll back, undo, and redo transactions in your deployment.

Rollback

The rollback feature undoes all transactions up to the point of the specified transaction.

Undo

The undo feature reverts a selected transaction only.

Redo

The redo feature repeats the recent transaction action.



Note: Active kernel cannot be rolled back for a kernel update.

Requirements

To use the YUM Transaction History dashboard, ensure that you have the following requirements.

- Use BigFix version 9.5 and later.
- Use Oracle Linux 6 and later versions.
- Use YUM version 3.2.28 and later.



Note: The rollback functionality is supported for the YUM versions 3.2.29 and later.



Note: In the dashboard, if the YUM version of a selected endpoint is earlier than version 3.2.29.22, a warning sign next to the YUM version will indicate that the version does not support the rollback action. The **Rollback** button is disabled for all transactions if the version is not supported.

- Subscribe to the Patching Support site.
- Activate the YUM Transaction History analysis.

YUM transaction actions

The **Actions** column identify the YUM transaction actions in the dashboard. The following table give details of the actions for every transaction.

Table 5. Description of transaction actions

Action	Abbre- viation	Description
Down- grade	D	At least one package has been downgraded to an older version.
Erase	E	At least one package has been removed.
Install	I	At least one new package has been installed.
Obsoleting	O	At least one package has been marked as obsolete.
Reinstall	R	At least one package has been reinstalled.
Update	U	At least one package has been updated to a newer version.

For more information about YUM history, go to [Red Hat Product Documentation site](#).

YUM transaction analyses

The dashboard uses the following analyses:

YUM Transaction History analysis

BigFix Patches for Oracle Linux generates a log which records the results of the actions that are taken in the YUM Transaction History dashboard. The YUM History Transaction analysis retrieves the content of the action log `yum_history.log`. The log is located in `/var/opt/BESClient/EDRDeployData/yum_history.log`.

YUM Logs analysis

YUM log is the official log that YUM generates by default in `/var/log/yum.log`. To change the default location, modify the log file settings in `/etc/yum.conf`. The YUM log analysis is very useful for troubleshooting purposes.

The analysis has 2 properties.

YUM Log property

Logs all the operations that are performed and identifies the transactions that are modified. This log retrieves the last 40 lines of the YUM log file.

YUM History Dashboard action log

This log lists the action logs for the redo, undo, and rollback operations from the action. The actions write the logs to `/var/opt/BESClient/EDRDeployData/yum_history.log`. The YUM History Dashboard action records the latest 5 actions.

Troubleshooting

To perform troubleshooting for the YUM Transaction History dashboard, you can check the `yum_history.log` file in `var/opt/BESClient/EDRDeployData`.

Rolling back a YUM transaction

Learn how to rollback YUM transactions.

Ensure that you meet the following requirements:

- Use BigFix version 9.5 and later.
- Use Oracle Linux 6 and later versions.
- Use YUM version 3.2.28 and later. The rollback functionality is supported for YUM version 3.2.29 and later.



Note: In the dashboard, if the YUM version of a selected endpoint is earlier than version 3.2.29.22, a warning sign next to the YUM version will indicate that the version does not support the rollback action. The **Rollback** button is disabled for all transactions if the version is not supported.

- Subscribe to the Patching Support site.
- Activate the YUM Transaction History analysis.

1. Using the BigFix console, go to **External Sites > Patching Support > Dashboards > YUM Transaction History**.
2. Select the endpoint whose YUM history you want to view.



Note: If the YUM version of a selected endpoint is earlier than version 3.2.29.22, a tooltip will indicate that the version is not supported.

3. Select the transaction that you want to roll back to.
4. Click **Rollback**.
5. The **Rollback Up To Transaction** window opens. OPTION: You can add flags in the field. Click **Apply**.
6. From the **Take Action** window, select the computer and click **OK** to run the action.

Undoing a YUM transaction

Use this feature to revert to a single, specific transaction.

Ensure that you meet the following requirements:

- Use BigFix version 9.5 and later.
- Use Oracle Linux 6 and later versions.
- Use YUM version 3.2.28 and later.



Note: The rollback functionality is supported for YUM version 3.2.29 and later.



Note: In the dashboard, if the YUM version of a selected endpoint is earlier than version 3.2.29.22, a warning sign next to the YUM version will indicate that the version does not support the rollback action. The **Undo** button is disabled for all transactions if the version is not supported.

- Subscribe to the Patching Support site.
- Activate the YUM Transaction History analysis.

1. Using the BigFix console, go to **External Sites > Patching Support > Dashboards > YUM Transaction History**.
2. Select the endpoint whose YUM history you want to view.
3. Select the transaction whose rollback that you want to undo.
4. Click **Undo**.
5. In the **Undo Transaction** window, click **Apply**.
6. From the **Take Action** window, select the computer and click **OK** to run the action.

Redo a YUM transaction

Use this feature to repeat the recent transaction action.

Ensure that you meet the following requirements:

- Use BigFix version 9.5 and later.
- Use Oracle Linux 6 and later versions.
- Use YUM version 3.2.28 and later.



Note: In the dashboard, if the YUM version of a selected endpoint is earlier than version 3.2.29.22, a warning sign next to the YUM version will indicate that the version does not support the rollback action. The **Redo** button is disabled for all transactions if the version is not supported.

- Subscribe to the Patching Support site.
- Activate the YUM Transaction History analysis.



Note: Active kernel cannot be rolled back for a kernel update.

1. Using the BigFix console, go to **External Sites > Patching Support > Dashboards > YUM Transaction History**.
2. Select the endpoint whose YUM history you want to view.
3. Select the transaction that you want to redo.
4. Click **Redo**.
5. In the **Redo Transaction** window, click **Apply**.
6. From the **Take Action** window, select the computer and click **OK** to run the action.

Checking for YUM package updates

You can use a task to identify YUM package updates that you must install in your deployment.

You can refer to a list that details the YUM package updates that are applicable to your Oracle Linux endpoints.

When you run the ID 39: YUM: Check Available Package Updates task, the YUM Logs analysis shows the results in a new column. The task uses the yum check-update to determine which updates are available for your installed packages. The task uses the repository in your deployment. Ensure that the YUM packages are available in your repository when you run the task.

Using the YUM: Check Available Package Updates task

- Subscribe to the Patching Support site.
 - Ensure that the YUM Logs analysis is activated.
 - Ensure that your endpoint uses a repository and that the YUM packages are available in the repository.
1. From the console, go to **Patching Support** site, select the following task: ID 39: YUM: Check Available Package Updates.
 2. Click **Take Action** to run the task.
 3. Click **OK**.
 4. When the action completes, go to **Patching Support > Analyses > Analysis: YUM Logs** and select the **Results** tab.

The YUM check-update output column is added and you can check if your endpoints have updates that must be installed.

Chapter 6. Manage custom repositories

Use the Oracle Linux Custom Repository Management dashboard to register and manage repositories.

Using custom repositories can give you the flexibility to control what can be deployed to the endpoints in your deployment. For example, you can deploy custom software that you are hosting in your custom repositories. Use the Oracle Linux Custom Repository Management dashboard to register and manage standard repositories and satellite repositories. Users of BigFix Patch for Oracle Linux can use the dashboard to do the following actions.

- Register, unregister, add, delete, and import custom repositories.
- Deliver custom software through BigFix. Support for custom repositories leverages on existing local repositories to save bandwidth and improve performance.

Instead of following the current BigFix infrastructure which allows Fixlets in the Patch for Oracle Linux site to directly download the patches from the Oracle Linux servers, the Fixlets now allow YUM to download from local repositories.

Ensure that you have met the following requirements.

- BigFix version 9.5 and later.
- Minimum YUM version: YUM 3.2.19-18
- Subscribe to the Patching Support site.
- Activate the Repository Configuration - Oracle Linux analysis from the Patching Support site to access the dashboard.



Note: The dashboard does not support the creation and maintenance of local repositories. You must create local repositories separately.

Use the Install packages by using the YUM task from the Patching Support site to install custom software that are in your custom repositories.

Registering a repository

Using the dashboard, you can register and connect your existing repositories to endpoints.

Activate the Repository Configuration - Oracle Linux analysis.

1. From the All Content domain, go to **Sites > External Sites > Patching Support > Dashboards > Oracle Linux Custom Repository Management**.
2. Click the **Endpoints** tab and select an endpoint. The repositories of the selected endpoints are listed in the lower part of the window. When a repository is named as unspecified, it means that it is not listed in the Repository list.
3. Click **Register a new repository**.
4. From the Register a New Repository window, select the repository, then click **Next**. The next window shows the name and the URL of the repository that you are registering.

5. This step is optional. You can also add more configuration information in **Additional Fields**. This information is saved in the YUM configuration files.



Note: Users who have custom repositories that are not just mirrors of the vendor sites must add `gpgcheck=0` in **Additional Fields**. When the gpg signature files are excluded, the rpm files are not checked for authenticity and might cause the installation to fail.

6. Click **Save**.
7. From the **Take Action** window, select the computer and click **OK** to run the action.

Unregistering a repository from an endpoint

Using the dashboard, you can unregister a repository from an endpoint.

When you unregister a repository, the dashboard removes the system ID file from the computer you selected. You must log in to the satellite server and delete the computer manually.

1. From the All Content domain, go to **Sites > External Sites > Patching Support > Dashboards > Oracle Linux Custom Repository Management**.
2. Click the Endpoints tab and click **Unregister a new repository**.
3. From the **Unregister a New Repository** window, select the repository and click **Save**.
4. From the Take Action window, select the computer and click **OK**.



Note: When you unregister a repository, the YUM configuration file is not deleted, but just disabled.

Adding repositories

You can add repositories with the Oracle Linux Custom Repository Management dashboard.

1. From the All Content domain, go to **Sites > External Sites > Patching Support > Dashboards > Oracle Linux Custom Repository Management**.
2. From the **Repositories** tab, select the repository that you want to add and click **Add**.
3. From the Add a New Repository window, enter values for the Name and Repository URL fields. **Note:** When you enter the satellite URL, the bootstrap URL is entered automatically. Bootstraps are created on the satellite server.
4. Click **Save**.

You added a repository to the dashboard. To have an endpoint use the repository that you added, go to the **Endpoints** tab and register the repository.

Importing a repository

Users can import their existing repositories using this feature.

Activate the Repository Configuration - Oracle Linux analysis to populate the dashboard with endpoint and repository information.

When importing existing repositories, ensure that the following entries are entered in the following order in the .repo files.

```
name=  
baseurl=  
enabled=  
gpgcheck=
```

1. From the All Content domain, go to **Sites > External Sites > Patching Support > Dashboards > Oracle Linux Custom Repository Management**.
2. Click the Repositories tab and click **Import**.
3. From the **Import Existing Repositories** window, select and name the repository.
4. Click **Save**.

The repository is now imported and added to the list of repositories in the dashboard.

Chapter 7. Using the download plug-in

The download plug-ins and OEL Plug-in are executable programs that downloads relevant packages directly from the patch vendor. Fixlets use an internal protocol to communicate with the download plug-in to download files. These Fixlets are based on updates made by the vendor.

For the Fixlet to be able to use the protocol, register the download plug-in on the BigFix server. Use the Manage Download Plug-ins dashboard to register the appropriate plug-in.



Notes:

- Download plug-ins support basic authentication only.
- The BigFix server and the BigFix client must be on the same version to avoid a null error.

Table 6. OEL Download Plug-in

Download Plug-in Name	Applicable Sites
OEL Plug-in	Patches for Oracle Linux 9

The **OEL Plug-in** downloads and caches patches directly from the vendor's website to the BigFix server, improving the accuracy and reliability of package dependency resolution and repository support.



Note: The **OEL Plug-in** does not work when the **Require SHA-256 Downloads** option in the BigFix Administration tool is enabled. When this option is enabled, all download verification use only the SHA-256 algorithm. However, there are certain repository metadata from the vendor, which do not contain SHA-256 values for packages in the repository that are used by the plug-in.

Consider disabling the **Require SHA-256 Downloads** option to successfully deploy a patch. Security and package integrity is not compromised as another layer of checking and verification is done using the GPG signature of the package. For more information about the download option, see BigFix Platform Installation Guide at https://help.hcltechsw.com/bigfix/9.5/platform/Platform/Installation/c_security_settings.html.

The download plug-in is highly extensible and robust, enabling such possibilities:

- Customize available repositories through a user extensible repository list.
- Installation and dependency resolution can easily be extended to all repositories, not just those that are shipped out of the box.
- Functionalities can easily be extended by customers and service teams.

It also improves performance related to downloading large numbers of packages, which consequently shortens the turnaround time for patching.

The revised OEL Plug-in access the following URL, refer to <https://yum.oracle.com/repo/OracleLinux/OL9>.

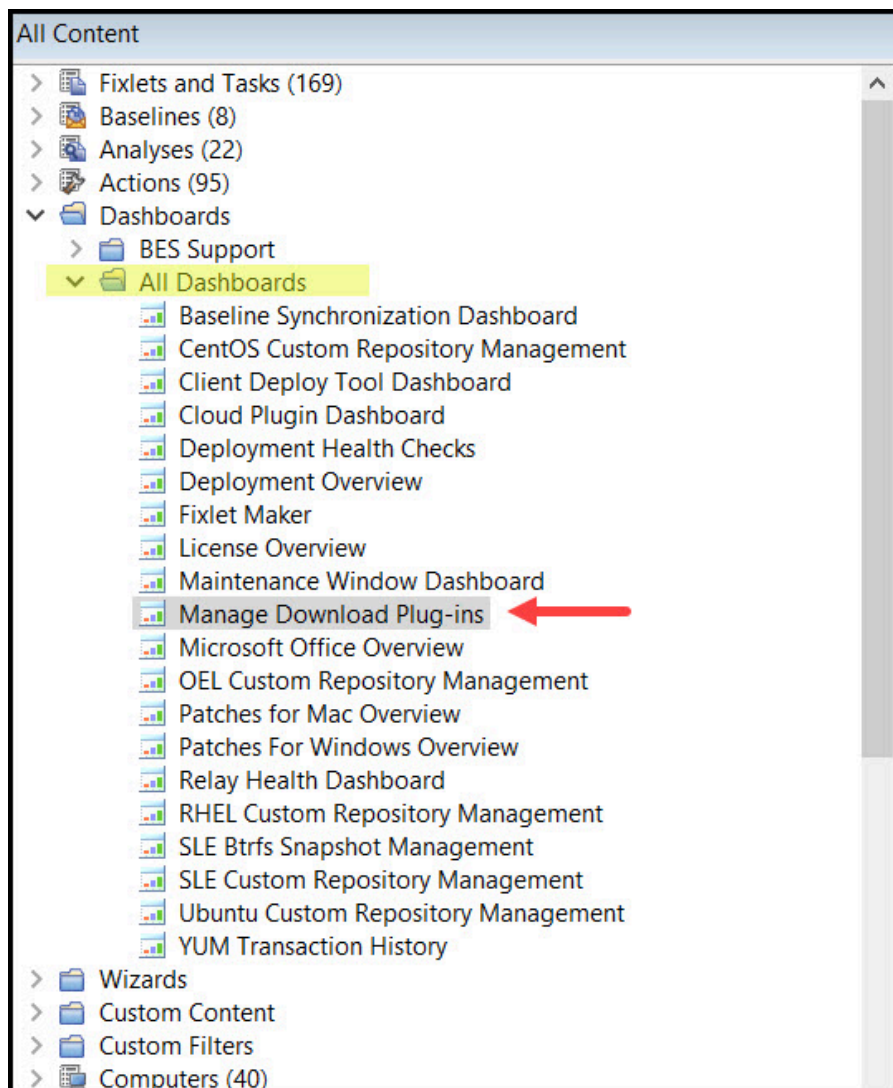
Manage Download Plug-ins dashboard overview

Use the Manage Download Plug-ins dashboard to oversee and manage download plug-ins in your deployment.

You can use the Manage Download Plug-ins dashboard to register, unregister, configure, and upgrade the download plug-ins for different patch vendors.

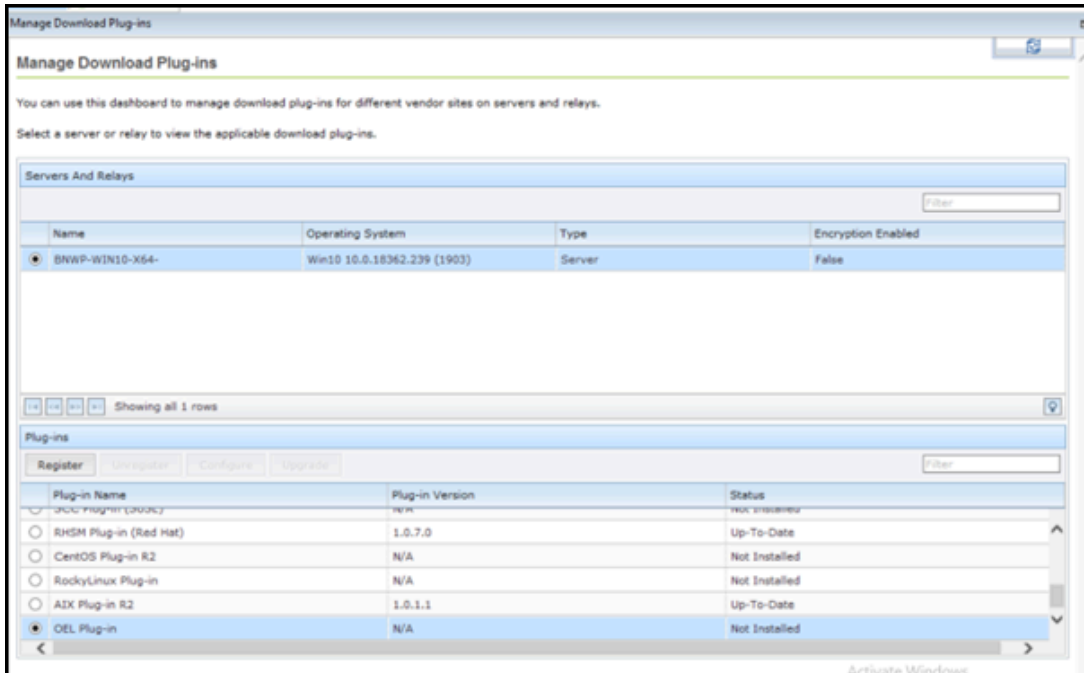
You must subscribe to the Patching Support site to gain access to this dashboard. To view the Manage Download Plug-ins dashboard, go to **All Content domain > All Dashboards > Manage Download Plug-ins**.

Figure 1. Patch Management navigation tree



The dashboard displays all the servers and windows-only relays in your deployment. Select a server or relay to view all the plug-ins for that computer. The dashboard shows you also the version and status for each plug-in in one consolidated view.

Figure 2. Manage Download Plug-ins dashboard



A plug-in can be in one of the following states:

- Not Installed
- New Version Available
- Up-To-Date
- Not Supported

The dashboard has a live keyword search capability. You can search based on the naming convention of the servers, relays, and plug-ins.



Note: If you install the download plug-in on BigFix relays, you must also install it on the BigFix server to avoid download issues.

Registering the Oracle linux Download Plug-in

Use the Manage Download Plug-ins dashboard to register the OEL Download Plug to install OEL patches from the Patches for Oracle Linux 9 sites.

You must complete the following tasks:

- Ensure that the BigFix server and the BigFix client are on the same version to avoid a null error.
- Subscribe to the **Patching Support** site to gain access to the Manage Download Plug-ins dashboard.
- Activate the **Encryption Analysis for Clients** analysis, which is available from the **BES Support** site.

- Activate the **Download Plug-in Versions** analysis, which is available from the **Patching Support** site.
- If you want to encrypt endpoints, deploy the **Enable Encryption for Clients** Fixlet, which is available from the **BES Support** site.

When you register the download plug-in on a computer without the plug-in, the plug-in is automatically installed and the configuration file is created.

If a download plug-in is already installed on the computer, the configuration file is overwritten.

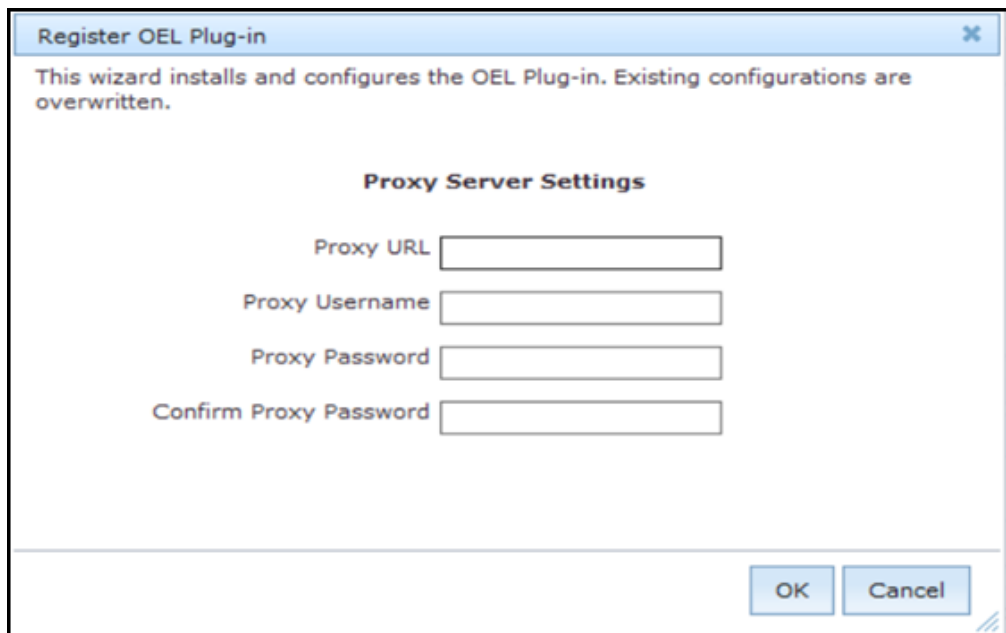
1. From the Patch Management domain, click **All Patch Management > Dashboards > Manage Download Plug-ins dashboard**.
2. From the Servers and Relays table, select the server on which the download plug-in is to be registered.

! **Important:** You must always register the download plug-in on the BigFix server.

3. From the Plug-ins table, select **OEL Plug-in**.
4. Click **Register**.

The Register OEL Plug-in wizard displays.

Figure 3. Register OEL Download Plug-in wizard



Register OEL Plug-in

This wizard installs and configures the OEL Plug-in. Existing configurations are overwritten.

Proxy Server Settings

Proxy URL

Proxy Username

Proxy Password

Confirm Proxy Password

OK Cancel

5. Enter the proxy parameters if the downloads must go through a proxy server.

Note: Only basic authentication is supported.

Proxy URL

The URL of your proxy server. It must be a well-formed URL, which contains a protocol and a host name. The URL is usually the IP address or DNS name of your proxy server and its port, which is separated by a colon. For example: `http://192.168.100.10:8080`.

Proxy Username

Your proxy user name if your proxy server requires authentication.

Proxy Password

Your proxy password if your proxy server requires authentication.

Confirm Proxy Password

Your proxy password for confirmation.

6. Click **OK**.
The Take Action dialog displays.
7. Select the target computer.
8. Click **OK**.

You successfully registered the OEL Download Plug-in.

Configuring the basic Oracle linux Download Plug-in settings

Use the Manage Download Plug-ins dashboard to configure the proxy settings of the OEL Plug-in.



Note: Ensure that the BigFix server and the BigFix client are on the same version to avoid a null error.

The scope of this task only covers the basic OEL Plug-in configuration from the BigFix console. To configure the advanced settings for the OEL Plug-in, you must edit the `plugin.ini` file. For details about how to do this, see [Configuring the advanced Oracle linux Download Plug-in settings \(on page 30\)](#).

You might want to take note of your existing configuration for the download plug-in. Existing configurations are overwritten when you configure the download plug-in.

1. From the Patch Management domain, click **All Patch Management > Dashboards > Manage Download Plug-ins dashboard**.
2. From the Servers and Relays table, select the server on which the download plug-in is to be configured.
3. From the Plug-ins table, select **OEL Plug-in**.
4. Click **Configure**.
The Configure OEL Plug-in wizard displays.

Figure 4. Configure OEL Download Plug-in wizard

Configure OEL Plug-in

This wizard configures the OEL Plug-in. Existing configurations are overwritten.

Proxy Server Settings

Proxy URL

Proxy Username

Proxy Password

Confirm Proxy Password

OK Cancel

5. Enter the proxy parameters if the downloads must go through a proxy server.



Note: Only basic authentication is supported.

Proxy URL

The URL of your proxy server. It must be a well-formed URL, which contains a protocol and a host name. The URL is usually the IP address or DNS name of your proxy server and its port, which is separated by a colon. For example: `http://192.168.100.10:8080`.

Proxy Username

Your proxy user name if your proxy server requires authentication.

Proxy Password

Your proxy password if your proxy server requires authentication.

Confirm Proxy Password

Your proxy password for confirmation.

6. Click **OK**.
The Take Action dialog displays.
7. Select the target computer.
8. Click **OK**.

You successfully configured the OEL Download Plug-in.

Configuring the advanced Oracle linux Download Plug-in settings

For advanced configurations, manually edit the OEL Download Plug-in configuration file called `plugin.ini`.

The `plugin.ini` file is automatically created when the download plug-in is registered from the Manage Download Plug-in dashboard. It contains the settings for logging and caching, as well as custom configurations for extending the repository list file.

On Linux systems, the file is in the root directory tree occupied by the download plug-in. For example, `/var/opt/BESServer/DownloadPlugins/OELProtocol`.

On Windows systems, the file is in the BigFix server installation directory. For example, `%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\OELProtocol`.

Figure 5. Example of the OEL Download Plug-in configuration file

```
[Logger]
file = logs/OELPlugin.log
level = INFO

[UA]
proxy =
proxyUser =
proxyPass =

primaryRepoListFile = C:\Program Files (x86)\BigFix Enterprise\BES Server\
    GatherDBData\gather\Patching Support\CurrentSiteData\DL0ELRepoList.json

extendedRepoListFile =
onlyUseExtendedRepoListFile = no

localCache =
localCacheOnly = no
```



Note: The `plugin.ini` is divided into sections, which are denoted by square brackets. Ensure that the options are under the correct sections. Moving the options to a different section might result in errors.

Either an absolute path or relative path can be used in the options that require a path: `file`, `primaryRepoListFile`, `extendedRepoListFile`, and `localCache`. Relative paths are relative to the download plug-in executable directory. By default, the executable file is in the `DownloadPlugins\OELProtocol` folder.

Setting the logging level

The logging level determines the amount of detail that is written to the `OELPlugin.log` file.

The available logging levels are as follows:

ERROR

Contains errors related to the execution of the download plug-in, which might indicate an impending fatal error.

WARNING

Contains information about failed downloads, and reasons for failure.

INFO

Contains general information outlining the progress and successful downloads, with minimal tracing information.

DEBUG

Contains fine-grained information used for troubleshooting issues. This is the most verbose level available.

You can change the logging level option from the `[Logger]` section of the `plugin.ini` file.

```
[Logger]
file = logs/OELPlugin.log
level = INFO
```

For example, if the logging is set to INFO, the logger outputs any logs for that level and any level above it. In this case, it outputs the INFO, WARNING, and ERROR logs.



Note: Setting the logging level to DEBUG increases the amount of information to log, which might impact performance. Only increase the logging level to DEBUG when investigating an issue, and switch back to INFO or WARNING after the issue is resolved.

Adding an extended repository list file

The OEL Download Plug-in can be configured to work with repositories that are not officially supported by BigFix, if required.

For more information about configuring the download plug-in to support such repositories, see [Extending the Oracle linux Download Plug-in \(on page 33\)](#).

Setting the download cache

You can use the download cacher tool to download the packages and repository metadata to a location that you specify.

There are three possible scenarios in which you can configure the download cacher for.

Sha1 download capability on air-gapped environments

The download cacher tool is mainly designed to be used for air-gapped environments, which require secure networks and therefore do not have access to the internet to download the files directly from the vendor site.

The sha1 download capability improves performance by caching the packages directly on the BigFix server's sha1 folder.

For information, see [Using the Oracle linux Download Cacher for air-gapped environments \(on page 42\)](#).

Sha1 download capability on an internet-enabled BigFix server

This method is considered best practice for caching packages on environments with a BigFix server that is internet-enabled. The sha1 download capability improves performance by caching the packages directly on the BigFix server's sha1 folder.

For information, see [Caching packages on the sha1 folder \(on page 43\)](#).

Without the sha1 download capability on an internet-enabled BigFix server

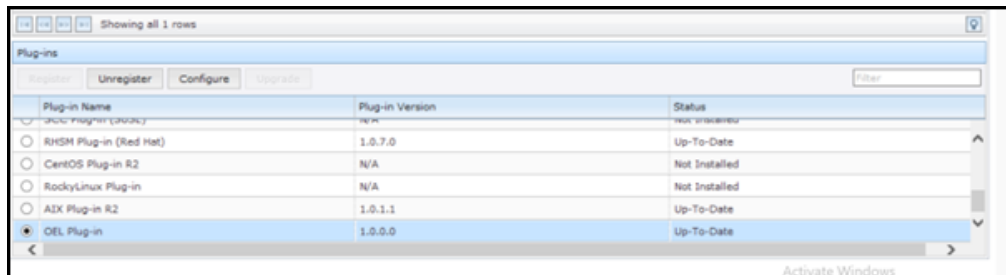
If for some reason you choose not to use the sha1 download capability to cache packages on the BigFix server's sha1 folder, you can use the local cache. For information, see [Caching packages on the local cache folder \(on page 44\)](#).

Unregistering the Oracle linux Download Plug-in

Use the Manage Download Plug-ins dashboard to unregister the OEL Download Plug-in.

1. From the Patch Management domain, click **All Patch Management > Dashboards > Manage Download Plug-ins dashboard**.
2. From the Servers and Relays table, select the server on which the download plug-in is to be unregistered.
3. From the Plug-ins table, select **OEL Plug-in**.
4. Click **Unregister**.

Figure 6. Unregister the OEL download plug-in



The Take Action dialog displays.

5. Select the target computer.
6. Click **OK**.

You successfully unregistered the OEL download plug-in.

Upgrading the Oracle linux Download Plug-in

Use the Manage Download Plug-ins dashboard to upgrade the OEL Download Plug-in to a later version.

1. From the Patch Management domain, click **All Patch Management > Dashboards > Manage Download Plug-ins dashboard**.
2. From the Servers and Relays table, select the server on which the download plug-in is to be upgraded.
3. From the Plug-ins table, select **OEL Plug-in**.
4. Click **Upgrade**.
The Take Action dialog displays.
5. Select the target computer.
6. Click **OK**.



Note: It is mandatory to re-configure the Download Plug-ins.



Note: The latest versions of Download Plug-ins are enhanced to strengthen the security of storing Proxy Password and Vendor Password.

You now have the latest version of the OEL Download Plug-in installed.

Extending the Oracle linux Download Plug-in

You can configure the OEL Download Plug-in to download and cache packages from repositories that are not officially supported by BigFix. The OEL Plug-in can resolve package dependency for unsupported OEL repositories, allowing the installation of unsupported dependent packages. This extended solution allows you to use BigFix to deploy packages from repositories that are not shipped out of the box.

Before you begin, you must complete the following tasks:

- Ensure that appropriate licenses are registered and that your OEL subscription is active.
- Review the BigFix Patch for OEL support information and verify the list of supported repositories:
 - [Supported architecture, erratas, and repositories \(on page 6\)](#)

To extend the plug-in to use repositories that are not officially supported by BigFix, you must carefully follow these steps:

1. [Create an extended repository list file \(on page 34\)](#).
2. [Update the OEL download plug-in R2 configuration file \(on page 35\)](#).
3. [Verify the extended repository setup \(on page 36\)](#).

Step 1: Creating an extended repository list file

The OEL Plug-in can use an extended repository list file, which contains additional repositories for both base and extension products that are included in your subscription.

The extended repository list file must meet the following criteria:

- It must be a valid `.json` file.
- It must be accessible to the OEL Plug-in. Ensure that the Administrator or root user can read the file.
- It must contain the following format:

```
{OEL Plug-in
  "<OS_key>": [
    {"name": "<name>", "baseurl": "<base_url>"},
    {"name": "<name>", "baseurl": "<base_url>"}
  ],
  "<OS_key_1>": [
    {"name": "<name>", "baseurl": "<base_url>"}
  ]
}
```



Note: It uses the same format as the `DLOELRepoList.json` file, which is set for `repoListFile` in the `plugin.ini` file.

You must replace the placeholders, which are enclosed in angle brackets `<>`, with the actual values.

OS_key

To use the unsupported repository with the BigFix OEL Patching content, use the following OS key as listed in the `DLOELRepoList.json` file:

- oel-9-x64

This might not be the full list as later versions of the download plug-in are released. To get the latest and complete copy of the repository list, complete the following actions:

1. Check whether your endpoints are registered to the latest download plug-in. The Manage Download Plug-ins dashboard indicates when the plug-in is up-to-date or when a new version is available.
2. View the `DLOELRepoList.json` file from the following locations:

On Windows Systems

```
%PROGRAM FILES%\BigFix Enterprise\BES Server\GatherDBData
\gather\Patching Support\CurrentSiteData
```

On Linux Systems

```
/var/opt/BESServer/gatherDBData/gather/Patching Support/
CurrentSiteData
```



Important: Use the correct OS key for each repository to avoid download and dependency resolution issues.

base_url

The base URL is formed from the OEL repository lists:

```
https://yum.oracle.com/repo/OracleLinux/<OracleOS_version>/<repo_name>/latest/<architecture>
```

```
https://yum.oracle.com/repo/OracleLinux/<OracleOS_version>/<repo_name>/<architecture>
```

Here is an example `.json` file with the prescribed format:

```
{
  "oel-9-x64": [
    { "name": "EPEL 7", "baseurl": "https://dl.fedoraproject.org/pub/epel/7/x86_64" }
  ]
}
```

We will name this file as `epel_repos.json` and use it in the examples throughout this section.

Step 2: Updating the OEL download plug-in configuration file

Configure the OEL Plug-in to use an extended repository list file. The `plugin.ini` configuration file is overwritten when the OEL Plug-in is unregistered or configured from the Manage Download Plug-in dashboard. Any change that you make in the configuration file is lost, therefore take note of the previous changes.

1. Use a text editor to open the `plugin.ini` file from the following locations:

On Windows systems

```
%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\OEL
Protocol
```

On Linux systems

```
/var/opt/BESServer/DownloadPlugins/OEL Protocol
```

2. In the **extendedRepoListFile** field, enter the absolute path or relative path to the extended repository list file. If it is set to a relative path, the path must be relative to the location of the OEL Plug-in executable.

For example:

```
extendedRepoListFile = ./epel_repos.json
```

This example indicates that the `epel_repos.json` file is located in the same directory as the download plug-in binary.

3. If you want set the OEL Plug-in to only use the extended repository list, set the **onlyUseExtendedRepoListFile** field to yes.

For example:

```
onlyUseExtendedRepoListFile= yes
```

If you want to set OEL Plug-in to use both repository list files, configure the setting to no. In cases where the same OS key is used in both files, the repositories will be combined.

4. Save the file.

Step 3: Verifying the extended repository setup

Use the OEL Download Cacher to verify whether the extended repository has been set up correctly and that you can it. For details about the cacher, see [Oracle linux Download Cacher usage information \(on page 37\)](#).

To do this, run the `OELDownloadCacher.exe --check-allrepos` command.

The expected output for a successful setup is as follows:

```
3488      : 2018-02-28 17:23:36 : INFO      : Testing access to: oel-6-x64
3488      : 2018-02-28 17:23:36 : INFO      : EPEL_6
3488      : 2018-02-28 17:23:36 : INFO      : Success!
3488      : 2018-02-28 17:23:36 : INFO      : OEL_-_OS
3488      : 2018-02-28 17:23:36 : INFO      : Success!

3488      : 2018-02-28 17:23:36 : INFO      : Testing access to: oel-9-x64
3488      : 2018-02-28 17:23:36 : INFO      : EPEL_7
3488      : 2018-02-28 17:23:36 : INFO      : Success!
3488      : 2018-02-28 17:23:36 : INFO      : OEL_-_OS
3488      : 2018-02-28 17:23:36 : INFO      : Success!
```

Chapter 8. Using the download cacher

The download cacher is a standalone command-line tool that is designed to download and cache files required for caching. The pre-cached files can be used by the download plug-in to patch the endpoints.

The download cacher is designed to be used for air-gapped environments. Use this tool to download and cache a large number of packages that are required by the Fixlets. By pre-caching the files, execution of actions are faster because you do not need to download the files from the Internet before distributing them to BigFix clients.



Note: If the BigFix server has access to the Internet, use the download plug-in. You must register the download plug-in from the Manage Download Plug-ins dashboard.

You can access the tool by downloading and running it manually.

Oracle linux Download Cacher usage information

Use the OEL Download Cacher to download and cache OEL patches in air-gapped environments. This tool supports the **Patches for Oracle Linux 9** sites.

The OEL Download Cacher accesses the <http://yum.oracle.com> to download the packages for the updates. To avoid download errors, ensure to configure your whitelist, firewall, or proxy to allow traffic through the listed hosts.

You can run the OEL Download Cacher on a Windows system or a Linux system. For information about requirements, see [BigFix 10.0 - System Requirements](#).

The latest OEL Download Cacher is available from the BigFix Support site:

- For Windows systems, download the tool at <http://software.bigfix.com/download/bes/util/OELDownloadCacher.exe>.
- For Linux systems, download the tool at <http://software.bigfix.com/download/bes/util/OELDownloadCacher-linux.tar.gz>. This tool is supported on x86-64 (64-bit) systems.



Note: To use the tool successfully, ensure to install the following packages and their dependencies:

- GLIBC version 2.2.5 or later
- GLIBC version 2.3 or later
- GNU/Linux kernel version 2.6.31 or later

For illustration purposes, this section indicates the steps to run the OEL Download Cacher in Windows. However, the parameters and subcommands to run the OEL Download Cacher are the same for both Windows and Linux systems.

You can run the tool `OELDownloadCacher.exe` to perform additional operations. To run this tool from the command prompt, use the following command:

```
OELDownloadCacher.exe [-h] [parameters...] {subcommand} [subparameters...]
```

where:

-h

Specifies the help message of a command instead of running the command.

parameters

Specifies the optional parameters to be used to configure the download cacher.

--proxyServer

Specifies the URL of the proxy server to use. It must be a well-formed URL that contains a protocol and a host name. The URL is usually the IP address or DNS name of your proxy server and its port, which is separated by a colon. For example: `http://192.168.100.10:8080`.

--proxyUser

Specifies the proxy user name if your proxy server requires authentication.

--proxyPass

Specifies the proxy password if your proxy server requires authentication.

Only basic authentication is supported.

--download_dir

Specifies the directory where the repository metadata files are cached.

If this parameter is not defined, the files are downloaded to the directory that is relative to the download cacher executable directory.

You can configure the OEL Download Plug-in to use the cached files by setting `localCache` in the `plugin.ini` file.

--sha1_download_dir

Specifies the directory where the packages are cached with a sha1 filename into a single flat directory. The cacher downloads all packages from all repositories (keys) as files in the specified directory.

Only the packages are stored in the `sha1_download_dir`. Each repository metadata is stored in the `download_dir`, and the OEL Repository directory structure is maintained.

Space-saving benchmarks are established with the use of the `--sha1_download_dir` through the `check-storagereq` subcommand. Using `--sha1_download_dir` have shown significant decrease in storage size, download size, and time when caching multiple repositories of the same OEL version. This is because many packages are duplicated among repositories with the same OEL version (for example, OEL-9-x64). Space is not saved if you only cache a single repository for each OEL version (for example, OEL-9-x64).



Note: When using this parameter, consider the cache limit of the BigFix server's sha1 file folder.

--redownload

Specifies the flag to re-download and overwrite existing RPM files that are in the download directory.

If this parameter is not defined, RPM files are not re-downloaded. However, metadata are, by default, downloaded and overwritten.

--verifyExistingPkgChecksum

Specifies the flag to enforce a checksum check for existing RPM files when trying to download packages using the "buildRepo", "downloadPkg", or "downloadbypatchid" subcommands.



Note: The checksum is set to 'off' by default.

--loglevel

Specifies the log level. You can choose among DEBUG', 'INFO', 'WARNING', or 'ERROR'. By default, the value is set to 'INFO'.

INFO

Contains general information outlining the progress and successful downloads, with minimal tracing information.

WARNING

Contains information about failed downloads, and reasons for failure.

ERROR

Contains errors related to the execution of the download plug-in, which might indicate an impending fatal error.

DEBUG

Contains fine-grained information used for troubleshooting issues. This is the most verbose level available.

--help

Specifies the full description and help of a command instead of running the command.

subcommand subparameter

Specifies the subcommand and subparameters to be used to run the download cacher.



Note: The subcommand and subparameter names are case-sensitive.

The *subparameter* varies for each *subcommand* as follows:

check-baserepos

Checks if the BigFix supported OEL base repositories can be accessed. The results are displayed in the command prompt and in the `<cache directory>\logs\OELDownloadCacher.log` file.

check-allrepos

Checks if the BigFix supported OEL base repositories and sub-repositories can be accessed. The results are displayed in the command prompt and in the `<cache directory>\logs\OELDownloadCacher.log` file.

check-storagereq

Checks the storage space requirement when using the `buildRepo` command with and without the `--sha1_download_dir` option. The results are displayed in the command prompt and in the `<cache directory>\logs\OELDownloadCacher.log` file.

showKeys

Outputs the list of OS keys for the supported repositories in the `<cache directory>\logs\OELDownloadCacher.log` file. An OS key indicates the OEL operating system version and architecture of a single OEL repository.

The syntax to run this subcommand is:

```
OELDownloadCacher.exe --download_dir <download_dir>
[parameters] showKeys
```

For example, `OELDownloadCacher.exe --download_dir C:\downloads showKeys`

buildRepo

Builds a local mirrored repository and downloads all the relevant files based on the specified OS key.

The syntax to run this subcommand is:

```
OELDownloadCacher.exe --download_dir <download_dir>
--sha1_download_dir <sha1_download_dir> [parameters]
buildRepo --key <OS_key1,OS_key2,...>
```

For example, `OELDownloadCacher.exe --download_dir C:\downloads --sha1_download_dir C:\sha1_downloads buildRepo --key OEL-9-x64`

where:

--key OS_key1,OS_key2,...

Specifies the OEL operating system version and architecture. Entries must be separated by a comma and must not include spaces. It must use the following format:

```
<product>-<version_number>-<architecture>
```

For example, `--key OEL-9-x64`.

downloadMetadataOnly

Downloads the metadata of the specified OS keys.

The syntax to run this subcommand is:

```
OELDownloadCacher.exe --download_dir <download_dir>
[parameters] downloadMetadataOnly --key <OS_key1,OS_key2,...>
```

For example:

```
OELDownloadCacher.exe --download_dir C:\downloads downloadMetadataOnly
--key oel-9-x64
```

where:

--key OS_key1,OS_key2,...

Specifies the OEL operating system version and architecture. Entries must be separated by a comma and must not include spaces. It must use the following format:

```
<product>-<version_number>-<architecture>
```

For example, `--key OEL-9-x64`.

downloadPkg

Downloads the listed RPM files for the specified OS key.



Note: If the package that you are downloading has dependencies, it is suggested that that buildrepo be used instead to avoid dependency issues

The syntax to run this subcommand is:

```
OELDownloadCacher.exe --download_dir <download_dir>
[parameters] downloadPkg --key <OS_key1,OS_key2...>
--pkg <pkg1,pkg2,...>
```

For example:

```
OELDownloadCacher.exe --download_dir C:\temp --redownload downloadPkg
--key oel-9-x64 --pkg web-assets-filesystem-5-15.e19.noarch.rpm
```

where:

--key *OS_key1,OS_key2,...*

Specifies the OEL operating system version and architecture. Entries must be separated by a comma and must not include spaces. It must use the following format:

```
<product>-<version_number>-<architecture>
```

For example, `--key web-assets-filesystem-5-15.e19.noarch.rpm`.

--pkg *pkg1,pkg2,...*

Indicates the package name.

Each entry must be separated by a comma and must not include spaces. For example, `--pkg web-assets-filesystem-5-15.e19.noarch.rpm, openblas-srpm-macros-2-11.e19.noarch.rpm`.

Using the Oracle linux Download Cacher for air-gapped environments

You can use the OEL Download Cacher for air-gapped environments by using the `buildRepo` subcommand to download all patches for a repository to a specified directory.

- Ensure you have access to the BigFix supported OEL base repositories and sub-repositories. To check, run the subcommand `check-allrepos`.
- Ensure you have enough space to download the repository metadata and packages. To check for the required storage space, run the subcommand `check-storagereq`.
- Increase the BigFix server's sha1 folder size limit by doing the following steps:
 1. From the BigFix console, right-click the computer and select **Edit Computer Setting**.
 2. Increase the **_BESGather_Download_CacheLimitMB** size.

The suggested size is the current BigFix server's sha1 folder size plus the size of `sha1_download_dir`.

If the size of the `sha1_download_dir` cannot be determined beforehand the suggested size of the `sha1_download_dir` is a minimum of 20GB per repository. Note that the minimum size of 20GB might increase over time.

1. Use the `buildRepo` subcommand with the `OELDownloadCacher.exe` file to download all the files for a repository to a specified directory. For example:

```
OELDownloadCacher.exe --download_dir C:\downloads
--sha1_download_dir C:\sha1_downloads
buildRepo --key OEL-9-x64
```

In this example, the download cacher tool downloads the packages to *sha1_download_dir* and the repository metadata to *download_dir* for "oel-9-x64" repository.

2. Transfer the *download_dir* to the air-gapped BigFix server and the sha1 files in *sha1_download_dir* to the BigFix Server's sha1 folder.
3. Open the configuration file of the OEL Download Cacher called `plugin.ini`. Configure the `plugin.ini` file with the following settings to set the BigFix Server to retrieve the repository metadata and required packages from the local cache instead of trying to retrieve them online.

```
localCache = <download_directory_specified_for_the_--download_dir_parameter>
localCacheOnly = yes
```

Caching packages on the sha1 folder

You can use the OEL Download Cacher to cache packages directly on an internet-enabled BigFix server's sha1 folder to improve performance during the Fixlet deployment.

- Ensure you have access to the BigFix supported OEL base repositories and sub-repositories. To check, run the subcommand `check-allrepos`.
- Ensure you have enough space to download the repository metadata and packages. To check for the required storage space, run the subcommand `check-storagereq`.
- Increase the BigFix server's sha1 folder size limit by doing the following steps:
 1. From the BigFix console, right-click the computer and select **Edit Computer Setting**.
 2. Increase the `_BESGather_Download_CacheLimitMB` size.

The suggested size is the current BigFix server's sha1 folder size plus the size of *sha1_download_dir*.

If the size of the *sha1_download_dir* cannot be determined beforehand the suggested size of the *sha1_download_dir* is a minimum of 20GB per repository. Note that the minimum size of 20GB might increase over time.

This task enables the BigFix Server to leverage the internet to ensure that the necessary files, such as the repository metadata and packages, are available during Fixlet deployment.

1. Use the `buildRepo` subcommand with the `OELDownloadCacher.exe` file to download packages from a repository to the BigFix server's sha1 folder. For example:

```
OELDownloadCacher.exe --download_dir C:\downloads --sha1_download_dir
C:\Program Files (x86)\BigFix Enterprise\BES Server\wwwrootbes\bfmirror\downloads\sha1
buildRepo --key oel-9-x64
```

In this example, the download cacher tool downloads the packages directly to the BigFix server's sha1 folder.

2. Open the configuration file of the OEL Download Cacher called `plugin.ini`. Configure the `plugin.ini` file with the following settings to set the BigFix Server to retrieve the packages from its sha1 folder.

```
localCache =
localCacheOnly = no
```

Do not specify a value for the `localCache` setting if the BigFix server is internet-enabled and the packages are saved directly in the BigFix server's sha1 folder.

The BigFix server will first check if the necessary files are in the download plug-in's cache and sha1 folders before retrieving them from the internet. If the repository metadata in the download plug-in's cache folder has expired, a new repository metadata is downloaded online. If the required packages do not exist in the BigFix server's sha1 folder, the packages are downloaded from the internet.

Caching packages on the local cache folder

You can use the OEL Download Cacher to cache packages on an internet-enabled BigFix server. You can configure the BigFix Server to use the repository metadata and packages from both the cache folder and the internet.

Store packages in a separate folder instead of the BigFix server's sha1 folder in case the folder size inflates. Since the BigFix server only stores the latest download, the stored packages might be replaced by newer files if the BigFix server sha1 folder size limit is too small.

Storing the packages in the local cache allows the OEL Download Plug-in to use it instead of getting it from the internet.

Additional space may be required as the package will also be cached in the BigFix server's sha1 folder when the BigFix server requests the package from the local cache.

1. Use the `buildRepo` subcommand with the `OELDownloadCacher.exe` file to download packages from a repository to the BigFix server's sha1 folder. For example:

```
OELDownloadCacher.exe --download_dir C:\downloads
buildRepo --key oel-9-x64
```

In this example, the download cacher tool downloads the packages to the specified download folder.

2. Open the configuration file of the OEL Download Cacher called `plugin.ini`. Configure the `plugin.ini` file with the following settings:

```
localCache = <location of the transferred download_dir>
localCacheOnly = no
```

With this setting, the BigFix server first checks the repository metadata in the download plug-in's cache folder. If it has not expired, the BigFix server will use that metadata. Otherwise, the BigFix server will get the repository metadata from the internet. As for the packages, the BigFix server initially checks if the packages exist in the sha1 folder, then proceeds to the localCache before it goes to the internet.

Chapter 9. Frequently asked questions

Learn the answers to frequently asked questions about Patch for Oracle Enterprise Linux.

Does Oracle Linux 9 uses Custom Repository Management dashboard and YUM Transaction History dashboard?

Yes, Custom Repository Management dashboard can be configured for Oracle Linux 9 and this enable the direct download also.

Does Oracle Linux use the download plug-in and the download cacher for Oracle Linux 6, Oracle Linux 7, and Oracle Linux 8?

No, download plug-in and the download cacher are not supported for Oracle Linux 6, Oracle Linux 7, and Oracle Linux 8. Only, Oracle Linux 9 uses the feature of the download plug-in and the download cacher. Users must configure Oracle's public repository for downlodng the patches for required packages.

Why are my Oracle Linux patch updates failing deployment?

Some repositories are not enabled by default as they might break Oracle Enterprise Linux upstream compatibility. This might cause patches to fail deployment and users might get the following similar error shown in the `EDRDeployData.log` file:

```
No package kernel-devel-3.-514.26.1.0.1.el7.x86_64 available.
No package kernel-headers-3.-514.26.1.0.1.el7.x86_64 available.
No package kernel-tools-3.-514.26.1.0.1.el7.x86_64 available.
No package kernel-tools-libs-3.-514.26.1.0.1.el7.x86_64 available.
No package perf-3.-514.26.1.0.1.el7.x86_64 available.
No package python-perf-3.-514.26.1.0.1.el7.x86_64 available.
Error: Nothing to do
```

Users need to decide if upstream compatibility is necessary.

Follow these steps to enable the missing repository:



Note: The following steps are applicable for all Oracle Linux version.

1. Go to `/etc/yum.repos.d/public-yum-ol7.repo` and search OL7 repository file.
2. If the entry for the repository in this case `[o17_MODRHCK]` exists, ensure that it is enabled.
3. If the repository entry does not exist, add the following entry:

```
[o17_MODRHCK]
name=Latest RHCK with fixes from Oracle for Oracle Linux $releasever ($basearch)
baseurl=http://yum.oracle.com/repo/OracleLinux/OL7/MODRHCK/\$basearch/
ggpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
gpgcheck=1
```

```
priority=20
enabled=1
```

4. Save your changes.

How to identify the packages belonging to a Oracle Linux repository?

Some repositories are not enabled by default and this might cause patches to fail deployment. There might be chance when the patches deployment failed for a package but users are not aware as which repository that package belongs to. In that case, use yum commands to identify the repository.

There are some basic commands available in yum, if the repository is added and enabled in `/etc/yum.repos.d/<somename>.repo`. Then using `yum info <pkgname>` will list the available packages. For example:

```
yum info cri-o-1.20.7-1.el8
Available Packages
Name       : cri-o
Version    : 1.20.7
Release    : 1.el8
Arch       : src
Size       : 9.6 M
Source     : None
Repo       : ol8_cloud_native_1.3
Summary    : Kubernetes Container Runtime Interface for OCI-based containers
URL        : https://github.com/cri-o/cri-o
License    : ASL 2.0
Description : CRI-O is meant to provide an integration path between OCI conformant runtimes and
the kubelet.
           Specifically, it implements the Kubelet Container Runtime Interface (CRI) using
OCI conformant runtimes.

Name       : cri-o
Version    : 1.20.7
Release    : 1.el8
Arch       : x86_64
Size       : 19 M
Source     : cri-o-1.20.7-1.el8.src.rpm
Repo       : ol8_cloud_native_1.3
Summary    : Kubernetes Container Runtime Interface for OCI-based containers
URL        : https://github.com/cri-o/cri-o
License    : ASL 2.0
```

Description : CRI-O is meant to provide an integration path between OCI conformant runtimes and the kubelet.

Specifically, it implements the Kubelet.

Do we need internet connectivity on the endpoints for deploying patches in Oracle Linux?

Yes, endpoints or BESClient directly downloads the required packages from configured Oracle's public repository accessing by internet. BES Server does not download the required patches.

If BESClient cannot access Internet, then set up a custom repository. Contact Oracle support to setup or create a custom repository.

"Oracle Linux Custom Repository Management" dashboard is used to register and manage custom repositories for Oracle Linux Clients.

What to do when Fixlets fail to install with the following message in the EDR log? "Warning: Nothing to install. Please check if you are using the latest kernel."

This message appears only in case of Fixlets that deploy kernel packages. A kernel Fixlet becomes relevant if the endpoint does not have the target kernel package installed or if the endpoint's active kernel is at a lower version than the target kernel package. An endpoint is still considered subject to kernel vulnerabilities even if it has the latest kernel installed but not using it actively.

To remediate the issue, restart the endpoint and ensure it is using the latest kernel available.

Appendix A. Support

For more information about this product, see the following resources:

- [BigFix Support Portal](#)
- [BigFix Developer](#)
- [BigFix Playlist on YouTube](#)
- [BigFix Tech Advisors channel on YouTube](#)
- [BigFix Forum](#)

Notices

This information was developed for products and services offered in the US.

HCL may not offer the products, services, or features discussed in this document in other countries. Consult your local HCL representative for information on the products and services currently available in your area. Any reference to an HCL product, program, or service is not intended to state or imply that only that HCL product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any HCL intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-HCL product, program, or service.

HCL may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

For license inquiries regarding double-byte character set (DBCS) information, contact the HCL Intellectual Property Department in your country or send inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

HCL TECHNOLOGIES LTD. PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. HCL may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-HCL websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this HCL product and use of those websites is at your own risk.

HCL may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

HCL

330 Potrero Ave.

Sunnyvale, CA 94085

USA

Attention: Office of the General Counsel

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by HCL under terms of the HCL Customer Agreement, HCL International Program License Agreement or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-HCL products was obtained from the suppliers of those products, their published announcements or other publicly available sources. HCL has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-HCL products. Questions on the capabilities of non-HCL products should be addressed to the suppliers of those products.

Statements regarding HCL's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to HCL, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. HCL, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS," without warranty of any kind. HCL shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).

Portions of this code are derived from HCL Ltd. Sample Programs.

Trademarks

HCL Technologies Ltd. and HCL Technologies Ltd. logo, and hcl.com are trademarks or registered trademarks of HCL Technologies Ltd., registered in many jurisdictions worldwide.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other product and service names might be trademarks of HCL or other companies.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the HCL website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of HCL.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of HCL.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

HCL reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by HCL, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

HCL MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.