BigFix
Patch for Alma Linux User Guide

Special notice

Before using this information and the product it supports, read the information in Notices *(on page xxxvi)*.

Edition notice

This edition applies to BigFix version 11 and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Overview

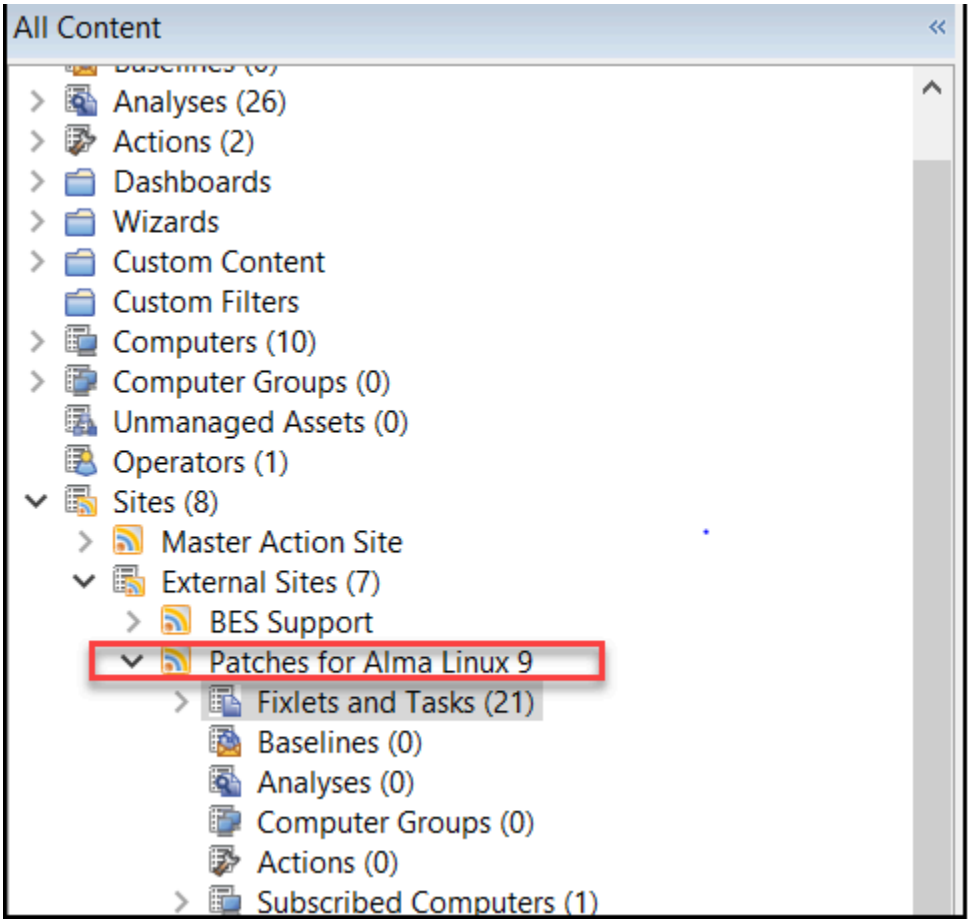
BigFix Patch for Alma Linux keeps your clients current with the latest updates and service packs.

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.

BigFix tests each Fixlet® in its laboratory before it is released and often finds issues that are dealt with by attaching extra notes to the Fixlet®. These notes typically allow the Console Operator to work around the problem, adding extra value to the patching process. BigFix incorporates also user feedback into notes, ensuring that you receive the latest information.

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.

Figure 1. Patch navigation tree



Supported platforms and updates

BigFix Patch for Alma Linux provides Fixlets for Alma Linux Security Advisories, Fix Advisories, and Enhancement Advisories on Alma Linux 9 platforms.

Table 1. Supported platforms and patches for BigFix Patch for Alma Linux

The following table lists the supported platforms and the corresponding sites that contains the Fixlets for patching.

Supported Platform	Fixlet Site Name	Type of Update
Alma Linux 9 (x86_64)	Patches for Alma Linux 9	See Supported Alma Linux repositories *(on page 7)* to view the list of repositories that contain the supported packages.

Supported Alma Linux repositories

BigFix Patch for Alma Linux supports the packages in several Alma Linux 9 repositories.

The following table lists the repositories that contain the supported packages for the different Alma Linux 9 versions.

Table 2. Supported Alma Linux repositories

Alma Linux version	Repository
Alma Linux 9.5 (x64)	Base: https://vault.almalinux.org/9.5/BaseOS/x86_64/os/ AppStream:https://vault.almalinux.org/9.5/AppStream/x86_64/os/ CodeReady Builder: https://vault.almalinux.org/9.5/CRB/x86_64/os
Alma Linux 9.6 (x64)	Base:https://repo.almalinux.org/almalinux/9.6/BaseOS/x86_64/os/ AppStream:https://repo.almalinux.org/almalinux/9.6/AppStream/x86_64/os/ CodeReady Builder: https://repo.almalinux.org/almalinux/9.6/CRB/x86_64/os/

Site subscription

Sites are collections of Fixlet messages that are created internally by you, by , 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 acquiring a Masthead file from a vendor or from or by using the Licensing Dashboard. For more information about subscribing to Fixlet sites, see the *Installation Guide*.

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

Chapter 2. Using the download plug-in for Alma Linux

The Alma Linux sites uses **CentOS Plug-in R2** for downloading the packages from the internet and caches it on BigFix server. CentOS Plug-in R2 is an executable download plug-in that retrieves the 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 3. Alma Linux Patches Download Plug-ins

Download Plug-in Name	Applicable Sites
CentOS Plug-in R2	Patches for Alma Linux 9

Note: Alma Linux supports only for version 9.5 and above. Versions earlier than 9.5 are not supported. To ensure access to patch content and repository updates, please upgrade your systems to Alma Linux 9.5 or later.

The **CentOS Plug-in R2** 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 **CentOS Plug-in R2** 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 Security.

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.

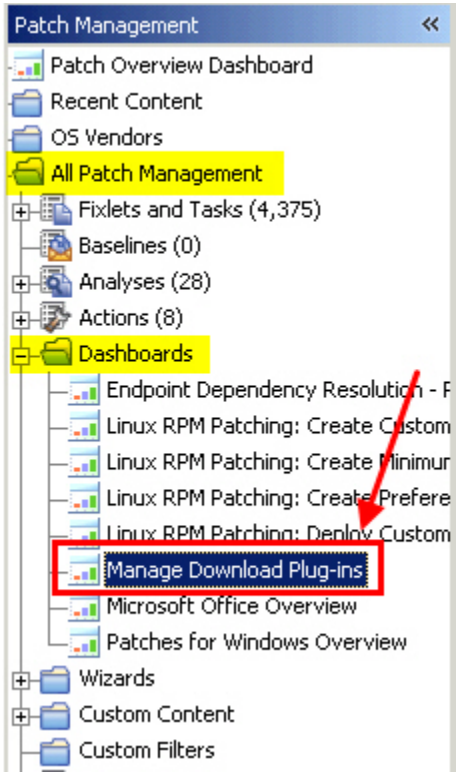
Manage Download Plug-ins dashboard overview

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.

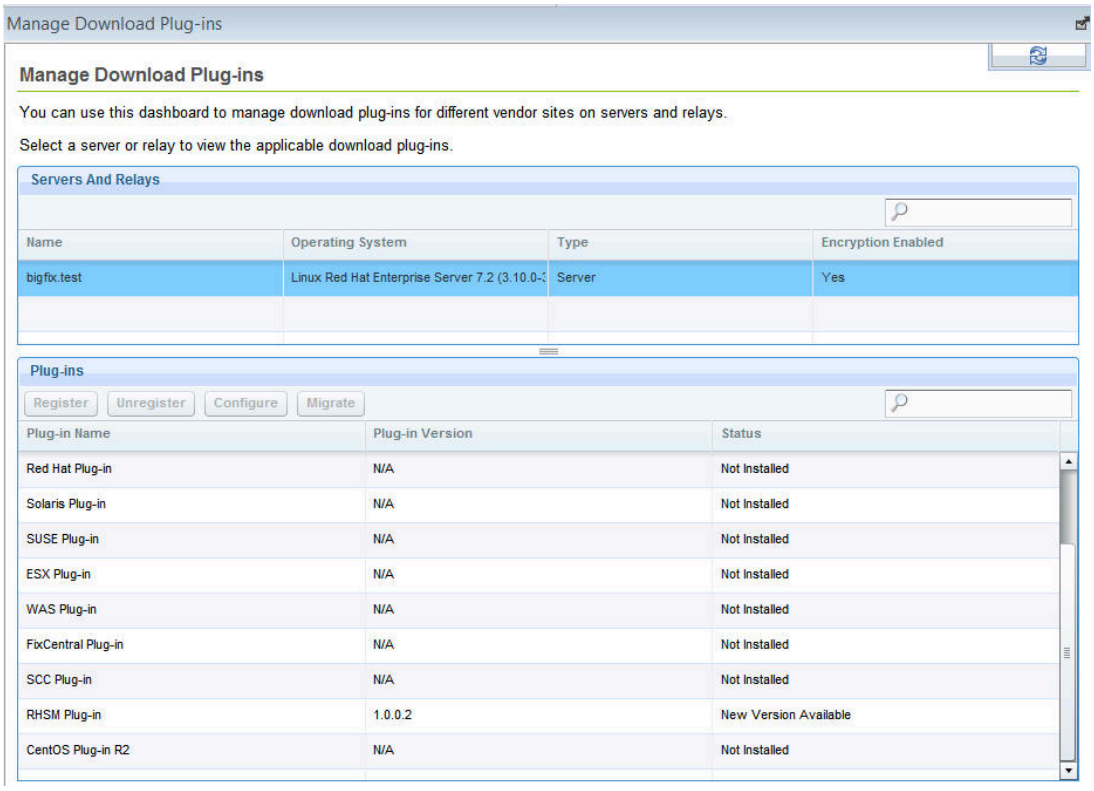
You must subscribe to the Patching Support site to gain access to this dashboard. To view the Manage Download Plug-ins dashboard, go to **Patch Management domain > All Patch Management > Dashboards > Manage Download Plug-ins**.

Figure 2. Patch Management



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 3. 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 Download Plug-in

To install the Alma Linux patches from their respective sites, register the CentOS Plug-in R2 download plug-in from the Manage Download Plug-ins Dashboard.

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.

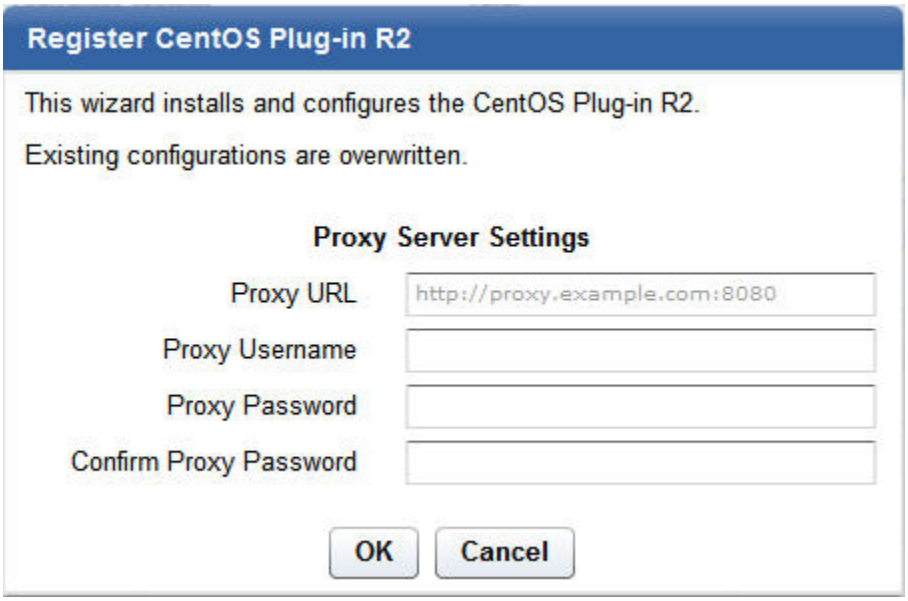
If a download plug-in is already installed on the computer, the configuration file is overwritten.

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 **CentOS Plug-in R2**.
4. Click **Register**.
   The Register CentOS Plug-in R2 wizard displays.

Figure 4. Register CentOS Download Plug-in R2 wizard



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 CentOS Plug-in R2 download plug-in.

Configuring Download Plug-in settings

Use the Manage Download Plug-ins dashboard to configure the proxy settings of the **Patches for Alma Linux**.

All the Alma Linux sites uses CentOS Plug-in R2 download plug-in from the Manage Download Plug-ins Dashboard. Refer to Configuring the basic CentOS Download Plug-in R2 settings to configure CentOS Plug-in R2 download plug-in.

Configuring the advanced Download Plug-in settings

All the Alma Linux sites uses CentOS Plug-in R2 download plug-in from the Manage Download Plug-ins Dashboard. Please refer to Configuring the advanced CentOS Download Plug-in R2 settings for configuring advanced settings.

Unregistering the Download Plug-in

All the Alma Linux sites uses CentOS Plug-in R2 download plug-in from the Manage Download Plug-ins Dashboard. Refer to Unregistering the CentOS Download Plug-in R2.

Upgrading the Download Plug-in

All the Alma Linux sites uses CentOS Plug-in R2 download plug-in from the Manage Download Plug-ins Dashboard. Refer to Upgrading the CentOS Download Plug-in R2.

Chapter 3. Using BigFix Patch for Alma Linux

BigFix Patch for Alma Linux enables automated patch management for Alma Linux systems. It uses Fixlets to detect and deploy relevant updates based on official Alma Linux repositories. The solution ensures consistent compliance and reduces manual effort by leveraging the BigFix platform to deliver, track, and report on patch status across Alma Linux 9.5 and later systems.

BigFix Patch provides a simplified patching process for downloading and installing patches that are relevant to a target endpoint. Use the Fixlets on the Patching Support and the various Alma Linux Fixlet sites to apply patches to your deployment and also you can view the Alma Linux bulletin for a specific Fixlet.

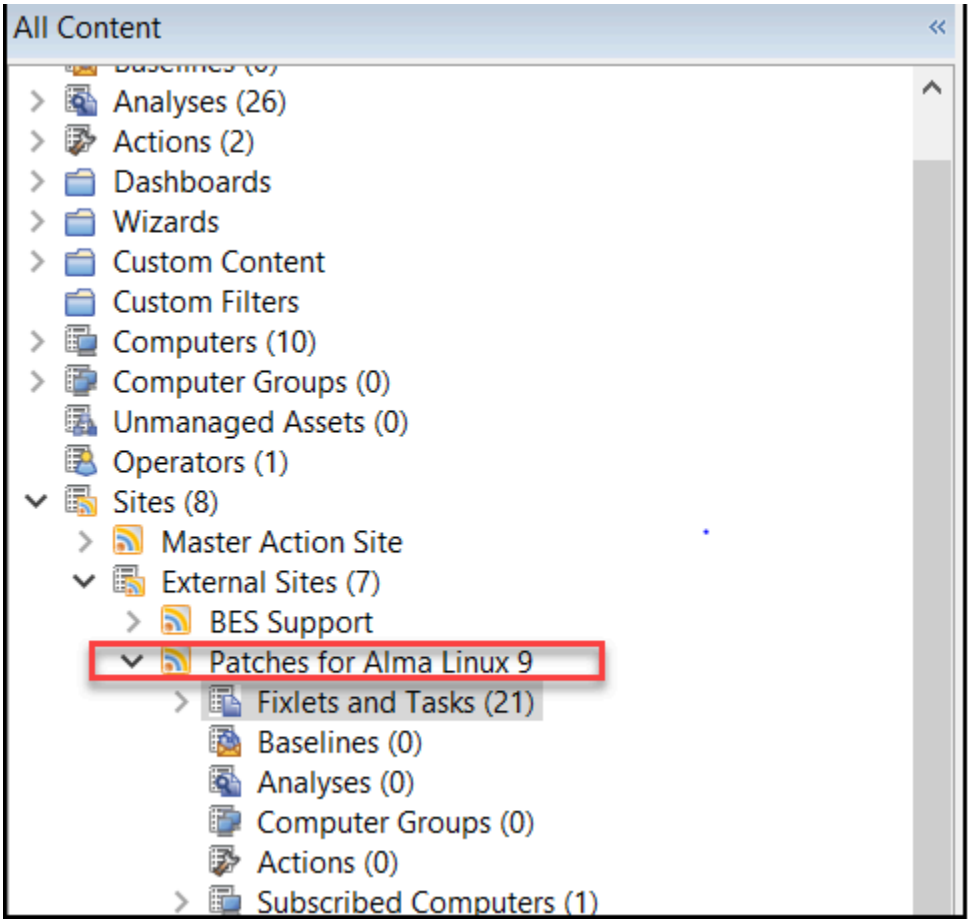
Patching using Fixlets

You can apply Alma Linux patches to your deployment by using the Fixlets that are available from the Alma Linux Fixlet sites.

Kernel Fixlets provide the option to upgrade or install all kernel packages. The upgrade option replaces existing kernel packages with later versions. The install option installs the later kernel packages next to the previous versions. The default behavior for kernel updates is to install packages side by side. Additionally, each kernel update Fixlet® provides the ability to test each of these options.

1. From the Patch Management domain, click **OS Vendors > Alma Linux**, and navigate to the patch content using the domain nodes.

Figure 5. Patch Management navigation tree

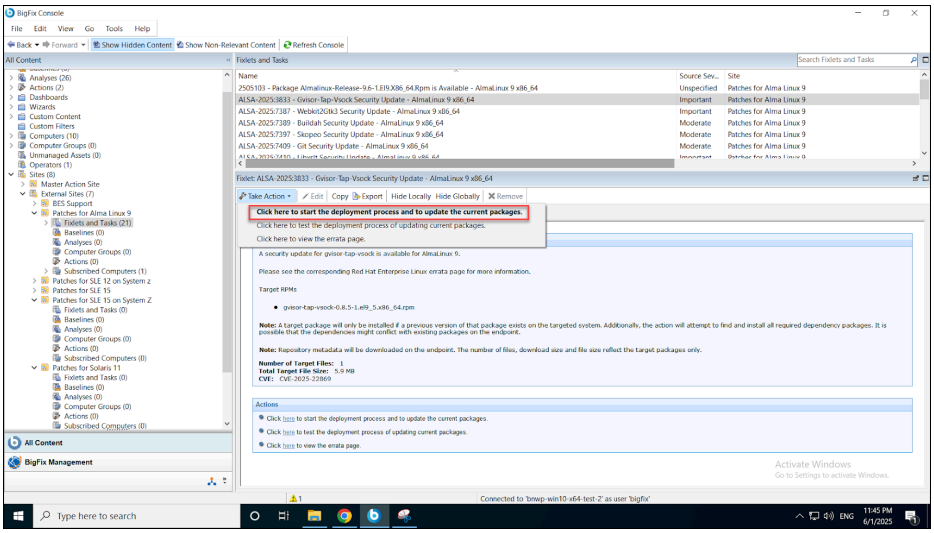


2. In the content that is displayed in the list panel, select the Fixlet that you want to deploy. The Fixlet opens in the work area.
3. Click the tabs at the top of the window to review details about the Fixlet.
4. Click **Take Action** to deploy the Fixlet.

   You can also click the appropriate link in the Actions list:

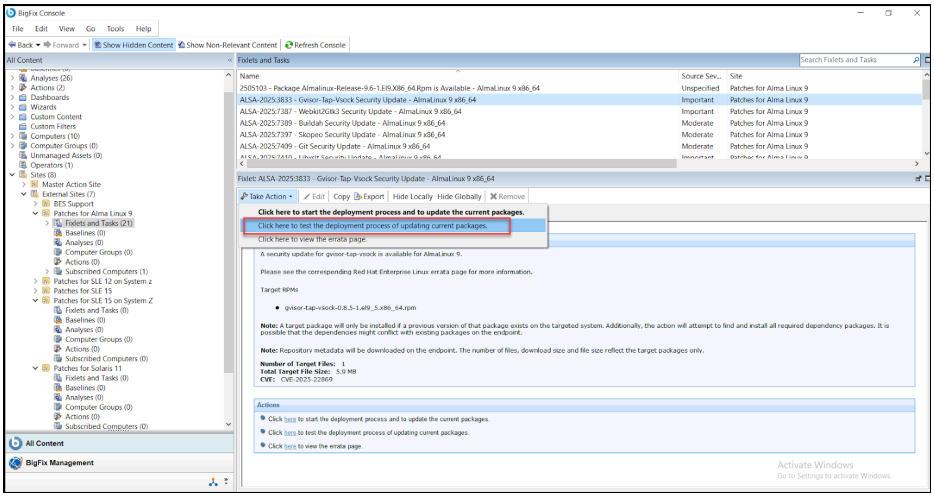
- You can start the deployment process.

  Figure 6. Take action to start deployment process

- You can deploy a test run prior to applying the patch. View the **Endpoint Dependency Resolution - Deployment Results** analysis to determine if the dependencies have been successfully resolved and if an installation is successful.

  Figure 7. Take action to deploy a test

- You can view the Security bulletin by selecting **Click here to view the patch page** action to view the patch page.

5. You can set more parameters in the Take Action dialog.

   For detailed information about setting parameters with the Take Action dialog, see the BigFix Console Operator's Guide.

6. Click **OK**.

Chapter 4. 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.

Alma Linux Download Cacher R2 usage information

Use the CentOS Download Cacher R2 to download and cache Alma Linux patches in air-gapped environments. This tool supports the Patches for Alma Linux 9 sites.

You can run the CentOS Download Cacher R2 on a Windows system or a Linux system. For information about requirements, see BigFix 11.0 - System Requirements.

The latest CentOS Download Cacher R2 is available from the BigFix Support site:

- For Windows systems, download the tool at https://software.bigfix.com/download/bes/util/CentOSR2DownloadCacher.exe.
- For Linux systems, download the tool at https://software.bigfix.com/download/bes/util/CentOSR2DownloadCacher-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 CentOS Download Cacher R2 in Windows. However, the parameters and subcommands to run the CentOS Download Cacher R2 are the same for both Windows and Linux systems.

You can run the tool `CentOSR2DownloadCacher.exe` to perform additional operations. To run this tool from the command prompt, use the following command:

```
CentOSR2DownloadCacher.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 CentOS Download Cacher R2 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 Alma Linux 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 Alma Linux version. This is because many packages are duplicated among repositories with the same Alma Linux version (for example, almalinux-9-x64). Space is not saved if you only cache a single repository for each Alma Linux version (for example, almalinux-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 Alma Linux base repositories can be accessed. The results are displayed in the command prompt and in the `<cacher directory>\logs\CentOSR2DownloadCacher.log` file.

check-allrepos

Checks if the BigFix supported Alma Linux base repositories and sub-repositories can be accessed. The results are displayed in the command prompt and in the `<cacher directory>\logs\CentOSR2DownloadCacher.log` file.

check-storagereq

Checks the storage space requirement when using the `builRepo` command with and without the `--sha1_download_dir` option. The results are displayed in the command prompt and in the `<cacher directory>\logs\CentOSR2DownloadCacher.log` file.

showKeys

Outputs the list of OS keys for the supported repositories in the `<cacher directory>\logs\CentOSR2DownloadCacher.log` file. An OS key indicates the Alma Linux operating system version and architecture of a single Alma Linux repository.

The syntax to run this subcommand is:

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
[parameters] showKeys
```

For example, `CentOSR2DownloadCacher.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:

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
--sha1_download_dir <sha1_download_dir> [parameters]
buildRepo --key <OS_key1,OS_key2,…>
```

For example, `CentOSR2DownloadCacher.exe --download_dir C:\downloads --sha1_download_dir C:\sha1_downloads buildRepo --key almalinux-9-x64`

where:

--key *OS_key1,OS_key2,…*

Specifies the Alma Linux 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 almalinux-9-x64`.

downloadMetadataOnly

Downloads the metadata of the specified OS keys.

The syntax to run this subcommand is:

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
[parameters] downloadMetadataOnly --key <OS_key1,OS_key2,…>
```

For example:

```
CentOSR2DownloadCacher.exe --download_dir C:\downloads downloadMetadataOnly
--key almalinux-9-x64
```

where:

--key *OS_key1,OS_key2,…*

Specifies the Alma Linux 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 almalinux-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:

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
[parameters] downloadPkg --key <OS_key1,OS_key2…>
--pkg <pkg1,pkg2,…>
```

For example:

```
CentOSR2DownloadCacher.exe --download_dir C:\temp --redownload downloadPkg
--key almalinux-9-x64 --pkg -pkg pki-symkey-11.0.6-2.el9_0.x86_64.rpm
```

where:

--key *OS_key1,OS_key2,…*

Specifies the Alma Linux 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 -pkg pki-symkey-11.0.6-2.el9_0.x86_64.rpm`.

--pkg *pkg1,pkg2,…*

Indicates the package name.

Each entry must be separated by a comma and must not include spaces. For example, `--pkg pki-symkey-11.0.6-2.el9_0.x86_64.rpm,glibc-langpack-wae-2.34-125.el9_5.3.alma.1.x86_64.rpm`.

Using Download Cacher for Air-Gapped Environments

You can use the CentOS Download Cacher R2 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 Alma Linux 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 `CentOSR2DownloadCacher.exe` file to download all the files for a repository to a specified directory. For example:

```
CentOSR2DownloadCacher.exe --download_dir C:\downloads
--sha1_download_dir C:\sha1_downloads
buildRepo --key almalinux-9-x64
```

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

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 CentOS Download Cacher R2 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 CentOS Download Cacher R2 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 Alma Linux 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 `CentOSR2DownloadCacher.exe` file to download packages from a repository to the BigFix server's sha1 folder. For example:

```
CentOSR2DownloadCacher.exe --download_dir C:\downloads --sha1_download_dir
C:\Program Files (x86)\BigFix Enterprise\BES Server\wwwrootbes\bfmirror\downloads\sha1
buildRepo --key almalinux-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 CentOS Download Cacher R2 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 CentOS Download Cacher R2 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 CentOS Download Cacher R2 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 `CentOSR2DownloadCacher.exe` file to download packages from a repository to the BigFix server's sha1 folder. For example:

   ```
   CentOSR2DownloadCacher.exe --download_dir C:\downloads
   buildRepo --key almalinux-9-x64
   ```

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

2. Open the configuration file of the CentOS Download Cacher R2 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 5. 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 YUM update transaction when the baseline is run. A single baseline can have numerous YUM calls, which can severely impact performance as it increases the time taken to complete all the transactions.

The multiple-package baseline installation solution helps address the poor 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 YUM 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.

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

A multiple-package installation task is made available for each Alma Linux operating system version 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.

Important: The **Enable the Multiple-Package Baseline Installation feature** and **Multiple-Package Baseline Installation** tasks must exist in the same baseline.

Available from the Patches for Alma Linux 9 site:

- Multiple-Package Baseline Installation - Alma Linux 9

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.

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.

`PkgToRemoveList.txt` file

This file contains the packages that are to be removed from the target Alma Linux 9 endpoint.

BigFix Patch also provides the following content to facilitate the installation:

Delete Alma Linux 9 Package List File for Multiple-Package Baseline Installation

Deletes the package list file on targeted Alma Linux 9 computers.

Add this task at the beginning of the baseline to avoid issues that might be related with resolving dependencies for the packages.

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.

- Set the client download precache size to at least 2GB. Use the `_BESClient_Download_PreCacheStageDiskLimitMB` setting to set the suggested size for the precache.
- Set the client download cache size to at least 2GB. Use the `_BESClient_Download_DownloadsCacheLimitMB` setting to set the suggested size for the cache.
- Ensure that you have at least 4GB of free disk space on each endpoint.
- Install and enable GPG keys on the endpoints. You can use the import release key task (ID# 301), available from the Fixlet sites, to import the necessary keys.

To run a single YUM call 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. For a list of the available tasks, see Multiple-Package Baseline Installation *(on page 24)*.

Notes:

- For Alma Linux 9, you need to sync all the Baselines before proceeding with installation because having superseded fixlets in the Baseline would lead to errors or installation failures. Only latest available packages are deployed with the Baseline.
- Packages with broken dependencies are skipped whenever possible. However, there are cases when packages cannot be skipped and the installation is canceled. These are some of the known cases:
  - Packages with dependency issues with Alma Linux.
  - Packages with dependency errors occur during installation, which is usually indicated by the following error message: `File conflicts happen when two packages attempt to install files with the same name but different contents`.
- Fixlets with `(vault x.y)` in the title will not work within multiple-package installation baselines.
- The multiple-package baseline installation feature does not support the pre-cache option **Start download before all constraints** are met in **Take Action**.

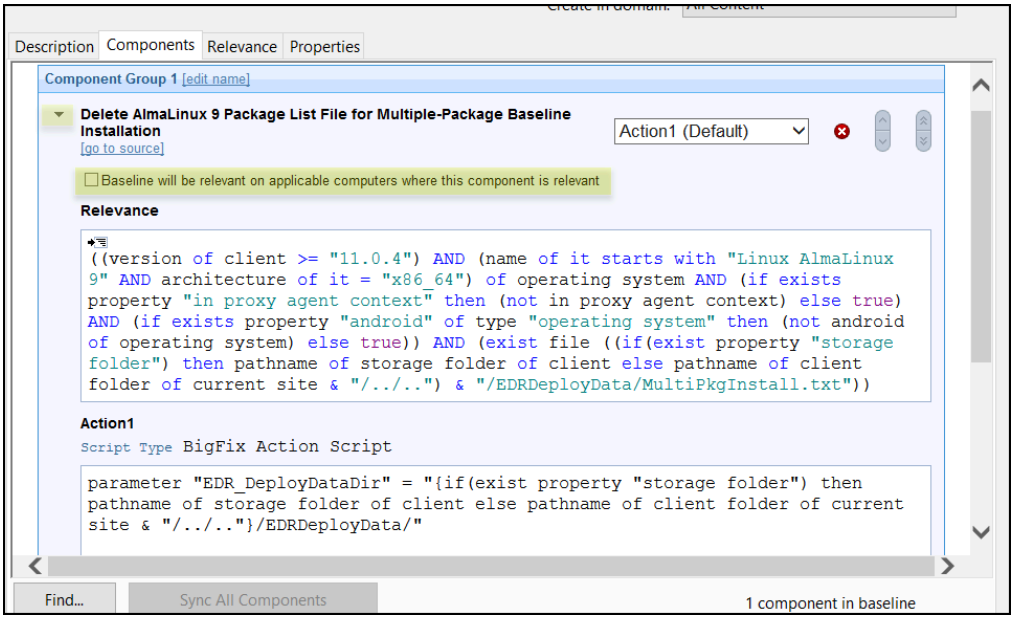
Important: The **Enable the Multiple-Package Baseline Installation feature** and **Multiple-Package Baseline Installation** tasks must be added in the same baseline in order for this feature to work. Also, the order of the tasks and Fixlets are crucial to the baseline installation, therefore take careful note of the steps as they must be taken in the correct order.

1. Create a baseline.

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

2. **Optional:** Add the **Delete Alma Linux 9 Package List File for Multiple-Package Baseline Installation** task before adding any of the Fixlets.
   Ensure that the **Baseline will be relevant on applicable computers where this component is relevant** option is not selected.

   Figure 8. Baseline component option



3. Add the appropriate **Enable the Multiple-Package Baseline Installation feature** task.
   Ensure that the **Baseline will be relevant on applicable computers where this component is relevant** option is not selected.
4. 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.

Note: If you add two or more Fixlets to the baseline that affect different versions of the same package, the installation task will skip the older versions of the package and install the latest one only.

5. 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 YUM transaction.

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

Before running the baseline, ensure that you meet the following requirements:

- The repositories that are registered on the endpoint must contain the target packages and all the required dependency packages.
- Allow enough time for a Fixlet, which is using the multiple-package installation method, to complete all YUM transactions and refresh the status on the endpoints before individually deploying the same Fixlet.
- Do not run multiple baselines from the same site on the same endpoint.
- Follow the Baseline Best Practices documented in the following technote: Baseline best practices

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 behavior. 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 6. YUM transaction management

View YUM transaction history and manage transactions through the **YUM Transaction History** dashboard, which works with Alma Linux.

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 YUM version 3.2.28 and later. The rollback functionality is not supported for the YUM versions 3.2.28 to 3.2.29-22.

   **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 4. Description of transaction actions

Action	Abbre-viation	Description
Down-grade	D	At least one package has been downgraded to an older ver-sion.

Table 4. Description of transaction actions (continued)

Action	Abbre-viation	Description
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 **YUM Transaction History** dashboard uses the following analyses:

YUM Transaction History analysis

BigFix 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 rollback capable field

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.

Rolling back a YUM transaction

Use the YUM Transaction History dashboard to roll back all transactions up to the point of the specified transaction.

Ensure that you meet the following requirements:

- Use YUM version 3.2.28 and later. The rollback functionality is not supported for the YUM versions 3.2.28 to 3.2.29-22.

  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.

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

1. From 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**.
   The **Rollback Up To Transaction** window opens.
5. **Optional:** You can enter additional flags to further customize the rollback action.
6. Click **Apply**.
7. From the **Take Action** window, select the computer and click **OK** to run the action.

Undo a YUM transaction

Use the YUM Transaction History dashboard to revert to a single, specific transaction.

Ensure you meet the following requirements:

- 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 **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.

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

1. From 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 The YUM Transaction History dashboard to repeat the recent transaction action.

Ensure that you meet the following requirements:

- 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.

Chapter 7. Custom Repositories Management

You can set up your custom repositories to manage patches for Alma Linux. This solution allows for multiple repositories on the entire deployment.

With the custom repository support, the Fixlets in the Alma Linux native tools sites can use DNF to directly download packages from custom repositories instead of going through the package sources at *http://repo.almalinux.org/* and *http://raw.repo.almalinux.org/*. Bandwidth throttling is not supported in a custom repository architecture.

Using custom repositories can give you the flexibility to control what can be deployed to the endpoints in your deployment.

Integrating your custom repository solution is made easy with the use of the **Alma Linux Custom Repository Management** dashboard. However, the dashboard does not add physical repositories; you must do this action separately.

Important: You must register the CentOS Plug-in R2 download plug-in when using custom repositories to avoid download errors. The download plug-in can be registered through the Manage Download Plug-ins dashboard.

Alma Linux Custom Repository Management dashboard

Use the Alma Linux Custom Repository Management dashboard to easily integrate your existing repository solutions with the BigFix patch management solution.

The Alma Linux Custom Repository Management dashboard allows Fixlets to use DNF for downloads instead of using the standard BigFix downloading infrastructure. The dashboard also allows you to register repositories to use the DNF commands when installing packages on the endpoints.

To access the dashboard, subscribe to the **Patching Support** site. From the Patch Management domain, click **All Patch Management > Dashboards > Alma Linux Custom Repository Management**. Activate the **Repository Configuration - Alma Linux** analysis to view the content in the dashboard.

Important: Your custom repositories must be pre-configured with the required metadata and headers before you use the dashboard.

Use the Alma Linux Custom Repository Management dashboard to perform the following actions for patch management:

- Register and unregister endpoints to a repository
- Add, delete, and import local repositories to the repository dashboard list

Note: The Alma Linux Custom Repository Management dashboard does not support the creation of a physical repository server. You must create the repository separately.

Important: You must register the CentOS Plug-in R2 download plug-in using custom repositories to avoid download errors. The CentOS Plug-in R2 download plug-in can be registered through the Manage Download Plug-ins dashboard.

Adding a repository

Use the Alma Linux Custom Repository Management dashboard to add a repository into the repository list so that you can register and connect it to endpoints.

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

1. From the **Alma Linux Custom Repository Management** dashboard, click the **Repositories** tab.
2. Click **Add**.
3. From the **Add a New Repository** dialog, enter values for the following fields:
    - **Repository Name**
    - **Repository URL**

    Note: Ensure that the repository settings match the repository server configuration.
4. Click **Save**.

To connect the added repository to an endpoint, see Registering Alma Linux endpoints to a repository *(on page 33)*.

If you want to add all the known existing repositories of an endpoint to the dashboard repository list, use the Import feature. For more information, see Importing repositories *(on page 35)*.

Registering Alma Linux endpoints to a repository

You can use the Alma Linux Custom Repository Management dashboard to add repositories as the package source when patching endpoints.

- Ensure that the repository settings match the repository server configuration.
- Activate the **Repository Configuration - Alma Linux** analysis if you have not yet done so.

1. From the **Alma Linux Custom Repository Management** dashboard, click the **Endpoints** tab.
2. Select the endpoints that you want to register to a repository from the first table. The repositories of the selected endpoints are listed in the second table.

Note: When a repository is named as unspecified, it means that it is not listed in the Repository list of the dashboard.

3. Click **Register a new repository**.
4. From the **Register a New Repository** dialog, select a repository and click **Next**.
   The next window shows the name and the URL of the repository that you are registering your endpoints to.
5. **Optional:** You can add more configuration information in **Additional Fields**.
   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**. This information is saved in the DNF configuration files.
7. From the Take Action dialog, select the computers and click **OK** to deploy the action.

You have successfully assigned the repository as the package source for the selected endpoints.

Run the **Enable custom repository support - Alma Linux** task to use the repository during patch deployment.

Unregistering Alma Linux endpoints from a repository

Use the Alma Linux Custom Repository Management dashboard to unregister endpoints from repositories that are no longer relevant.

When you unregister a repository, the DNF configuration file is not deleted, but just disabled. The Alma Linux Custom Repository Management dashboard also removes the system ID file from the computer that you selected.

1. From the **Alma Linux Custom Repository Management** dashboard, click the **Endpoints** tab.
2. Select the endpoints that you want to unregister a repository from.
3. Click **Unregister a new repository**.
4. From the **Unregister a New Repository** dialog, select a repository and click **Save**.
5. From the Take Action dialog, select the computers and click **OK** to deploy the action.

Deleting repositories

To manage the dashboard repository list more easily, delete the repositories that no longer exist in your deployment.

1. From the **Alma Linux Custom Repository Management** dashboard, click the **Repositories** tab.
2. Select the repositories that you want to delete and click **Delete**. A delete confirmation dialog displays.
3. Click **Yes** to confirm and proceed with the deletion of the selected repositories.

The selected repositories are removed from the list.

Importing repositories

Use the Import feature of the Alma Linux Custom Repository Management dashboard to add all the known existing repositories of an endpoint to the list of repositories in the dashboard.

Activate the **Repository Configuration - Alma 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` file:

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

1. From the **Alma Linux Custom Repository Management** dashboard, click the **Repositories** tab.
2. Click **Import**.
3. From the **Import Existing Repositories** dialog, select the repositories that you want to add in the dashboard repository list.
4. Enter a name for the repository.
5. Click **Save**.

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

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.