HCL

BigFix
Patch for CentOS User's Guide

Special notice

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

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 CentOS Linux™ keeps your Linux™ 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 Management navigation tree
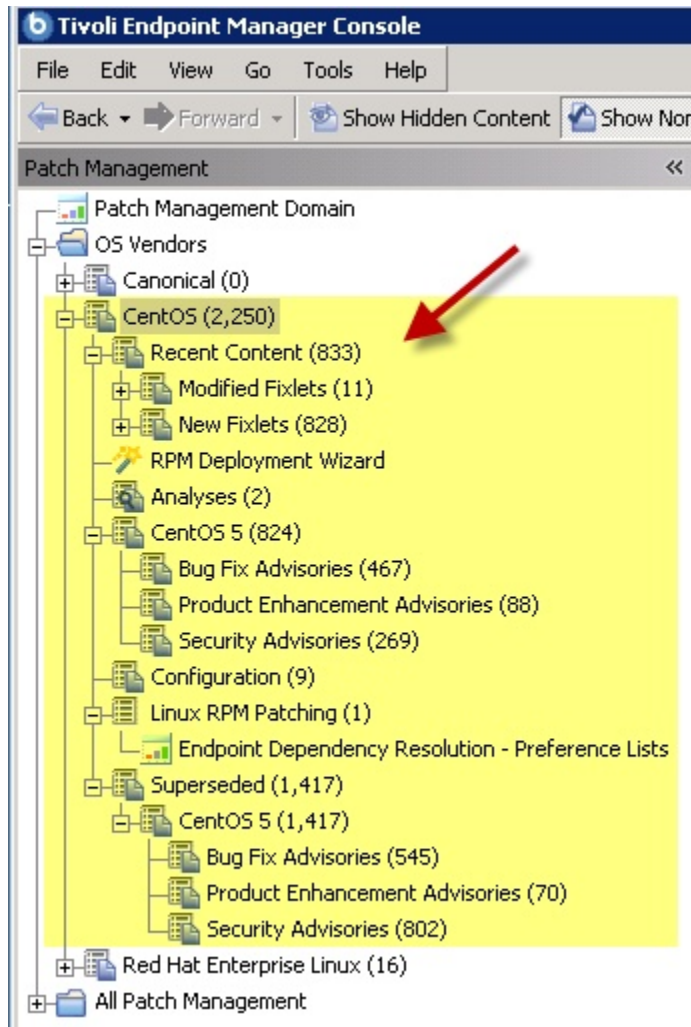
What's new in this update release

This release of BigFix Patch for CentOS contains various enhancements for patching.

CentOS 8 content support

The **CentOS Linux™ Enterprise 8 (x86_64)** content is published in a new site called **Patches for CentOS 8**.

This release supports the package manager's native command-line interface: DNF, which allows you to patch CentOS 8 systems. DNF is a wrapper on YUM and also reduces dependency issues, improves performance and is more reliable in terms of installing security patches.

CentOS Download Plug-in R2 enhancements

The CentOS Download Plug-in R2 v1.0.0.2 or later has the ability to use packages that are cached by the CentOS R2 Download Cacher's *download_dir* (referred to as localCache in the `plugin.ini` file) and get packages from the internet at the same time.

Previously, the CentOS Download Plug-in R2 was used in BigFIx server that was in either of the following scenarios:

- The BigFix Server is fully air-gapped and uses the CentOS Download Cacher R2, or
- The BigFix Server is internet-enabled and does not use the CentOS Download Cacher R2

With this enhancement, you can cache the packages offline to save time downloading the packages.

For more information, see .

CentOS Download Cacher R2 enhancements

The following enhancements are included in the CentOS Download Cacher R2 v1.0.0.2 or later:

Package sha1 download support

The CentOS Download Cacher R2 can now download packages as sha1 files instead of the RPM format using `--sha1_download_dir`.

Previously, when using the "buildRepo –key centos-7-x64" with the download cacher, the CentOS repository "centos-7-x64" structure is mirrored offline. This might result to duplication of packages if they are found in multiple repositories.

Using `--sha1_download_dir` will download all packages from all repositories (keys) as files with a sha1 filename into a single flat directory.

Repository access check

New commands to verify if you have access to the BigFix supported CentOS base repositories and sub-repositories: `check-baserepos` and `check-allrepos`

Storage space requirement check

New command to calculate and check the storage space requirement when using the builRepo command: `check-storagereq`

This command outputs the required space to download the repository metadata and packages with and without the use of the `--sha1_download_dir` option.

Space-saving benchmarks

Space-saving benchmarks have been established with the use of the `--sha1_download_dir` option.

Using the `--sha1_download_dir` option have shown significant decrease in storage size, download size, and time when caching multiple repositories of the same CentOS version. This is because many packages are duplicated among repositories with the same CentOS version (for example, centos-6.8-

x64, centos-6.7-x64, centos-6.6-x64). Space is not saved if you only cache a single repository for each CentOS version (for example, centos-6.8-x64, centos-7.1-x64).

For more information, see the following topics:

- CentOS Download Cacher R2 usage information *(on page 31)*
- Using the CentOS Download Cacher R2 for air-gapped environments *(on page 36)*
- Caching packages on the sha1 folder *(on page 37)*
- Caching packages on the local cache folder *(on page 38)*

Table 1. Previous updates

Enhancement or Feature	Description	Resources
Multiple-package baseline installation	BigFix Patch offers a solution that can combine the installation of updates for multiple packages into a single task, effectively reducing the execution time of the baseline. This solution is available on Fixlets from the Patches for CentOS6 Plugin R2 and Patches for CentOS7 Plugin R2 sites.	Multiple-Package Baseline Installation *(on page 46)*
Enhanced logging and error handling	Redesigned error logs and debug outputs to provide clearer error reporting to help reduce the time spent troubleshooting and debugging issues. This enhancement is available on Fixlets from the Patches for CentOS6 Plugin R2 and Patches for CentOS7 Plugin R2 sites.	Setting the logging level *(on page 23)*
Custom repository support	With the custom repository support, you can use YUM to download patches from the local repositories and distribute them to CentOS endpoints. You can also use your custom repositories to deliver custom software through BigFix.	Custom repositories management *(on page 51)*
YUM Transaction History dashboard	BigFix allows users to view all YUM-related transactions in a single dashboard to monitor the operations and commands that were run against the endpoints. This dashboard is also equipped with transaction management features such as rollback, undo, and redo.	YUM transaction management *(on page 57)*
CentOS 7 content support	The CentOS Linux™ Enterprise 7 (x86_64) content is published in a new site called **Patches for CentOS 7**. This release supports the package manager native command-line interface, YUM, which allows you to patch Cent-	Supported platforms and updates *(on page 11)* Patching method *(on page 14)*

Table 1. Previous updates (continued)

Enhancement or Feature	Description	Resources
	OS 7 systems. YUM reduces dependency issues, improves performance, and is more reliable in terms of installing security patches.	Use the RPM Deployment Wizard *(on page 42)*
	You must install the bzip2 utility on the endpoints to use the patches from the **Patches for CentOS 7** site. The bzip2 utility decompresses the metadata file that contains all the package information to the endpoints.	
	Note: The bzip2 utility might not be included in the CentOS 7 base installation, therefore manual installation of this compression utility is required.	
	Use the Fixlet named **Install bzip2** (ID #1) to identify the endpoints that require the installation of the bzip2 utility. Download the installation package from the vendor website and use the RPM Deployment Wizard, which is available from the Linux RPM Patching site, to deploy the package to various endpoints.	
CentOS 8 content support	The **CentOS Linux™ Enterprise 8 (x86_64)** content is published in a new site called **Patches for CentOS 8**.	**Patches for CentOS 8** site only supports BaseOS and AppStream repositories. For more information, see Supported CentOS Repositories
	This release supports the package manager's native command-line interface: DNF, which allows you to patch CentOS 8 systems. DNF is a wrapper on YUM which reduces dependence issues, improves performance and is more reliable in terms of installing security patches.	
	Announcements	
	CentOS 8 will not make announcements on its official site as in earlier versions. BigFix Patch team retrieves the package information from https://feeds.centos.org/ for generating the content based on the changes mentioned in BaseOS and AppStream change list. Red Hat API's are used to get all the dependent packages and errata information.	
	Supersedence	

Table 1. Previous updates (continued)

Enhancement or Feature	Description	Resources
	CentOS 8 repositories metadata now contains information related to latest available packages, hence all the superseded fixlets are expected to fail with an error `No matching Package found`. Once the fixlet is superseded, the fixlet is made non-relevant by default. The value(client) of _BESClient_CentOS_EnableSupersededEval is removed for superseded fixlets and relevance for superseded fixlets will be set to false `<Relevance>False</Relevance>`. Please deploy only the latest available patches. In order to deploy superseded fixlet using custom site, the relevant repository metadata and packages must be precedently cached. **Note:** You should sync all the Baselines before proceeding with installation because having superseded fixlets in the Baseline would lead to errors and installation failures. The latest available packages are deployed with the baseline.	

Supported platforms and updates

BigFix Patch for CentOS Linux™ provides Fixlets for CentOS Security Advisories, Fix Advisories, and Enhancement Advisories on CentOS Linux™ 6 and 7 platforms.

Table 2. Supported platforms and patches for BigFix Patch for CentOS

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
CentOS Linux 6 (x86, x86_64)	Patches for CentOS6 Plugin R2	See Supported CentOS repositories *(on page 13)* to view the list of repositories that contain the supported packages.
CentOS Linux 7 (x86_64)	Patches for CentOS7 Plugin R2	See Supported CentOS repositories *(on page 13)* to view the list of repositories that contain the supported packages.

Table 2. Supported platforms and patches for BigFix Patch for CentOS

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

(continued)

Supported Platform	Fixlet Site Name	Type of Update
CentOS Linux 8 (x86_64)	Patches for CentOS 8	See Supported CentOS repositories *(on page 13)* to view the list of repositories that contain the supported packages.

📝 **Notes:**

- **CentOS 8** has reached its End of Life on December 31, 2021 which means **CentOS** will no longer provide updates on **CentOS Linux 8**. Hence BigFix will not provide the Patch content for the same. We are planning to deprecate the site soon. For more information, see https://www.centos.org/centos-linux-eol/.
- **CentOS 8** Stream is not supported by BigFix Patch.
- Superseded fixlets what are part of **Patches for CentOS 8** site fails to install because the metadata of the CentOS 8 is missing older package information and also updates only to the latest version.
- **CentOS 8** has stopped publishing announcement information and hence all the fixlets are created based on the feeds published by CentOS.
- Fixlet content that have been superseded for over a year are removed from the external sites and archived into a different site. Contact your HCL Technical Sales Representative if you need access to the superseded content.
- The **Patches for CentOS 6 Native Tools** and **Patches for CentOS 7** sites have been deprecated on September 29, 2017. BigFix Patch no longer provide Fixlet content or support for these sites after the deprecation date.
- The **Patches for CentOS 5 Native Tools** site has been deprecated on July 31, 2017. BigFix Patch no longer provides content or support for CentOS Linux 5 after the deprecation date.
- The **Patches for CentOS 5** site and **Patches for CentOS 6** site have been deprecated on December 31, 2016. BigFix Patch no longer provides content and support for these sites.

BigFix generates Fixlet content based on the CentOS advisories published through the following announcement mailing lists:

CentOS-announce

CentOS announcements containing security and general information, including errata packages. To subscribe to this mailing list, go to https://lists.centos.org/mailman/listinfo/centos-announce.

CentOS-CR-announce

CentOS announcements regarding the CR repositories for various versions. To subscribe to this mailing list, go to https://lists.centos.org/mailman/listinfo/centos-cr-announce.

Important: BigFix does not generate content for packages that are released without an official CentOS advisory.

Supported CentOS repositories

BigFix Patch for CentOS supports the packages in several CentOS repositories.

The following table lists the repositories that contain the supported packages for the different CentOS versions.

Table 3. Supported CentOS repositories

CentOS version	Repository
CentOS 6 (x86)	Base: `http://mirror.centos.org/centos/6/os/i386/` Updates: `http://mirror.centos.org/centos/6/updates/i386/` Extras: `http://mirror.centos.org/centos/6/extras/i386/` Fasttrack: `http://mirror.centos.org/centos/6/fasttrack/i386/`
CentOS 6 (x86_64)	Base: `http://mirror.centos.org/centos/6/os/x86_64/` Updates: `http://mirror.centos.org/centos/6/updates/x86_64/` Extras: `http://mirror.centos.org/centos/6/extras/x86_64/` Fasttrack: `http://mirror.centos.org/centos/6/fasttrack/x86_64/`
CentOS 7 (x86_64)	Base: `http://mirror.centos.org/centos/7/os/x86_64/` Updates: `http://mirror.centos.org/centos/7/updates/x86_64/` Extras: `http://mirror.centos.org/centos/7/extras/x86_64/` Fasttrack: `http://mirror.centos.org/centos/7/fasttrack/x86_64/`
CentOS 8 (x86_64)	Base:http://mirror.centos.org/centos/8/BaseOS/x86_64/os/ AppStream:http://mirror.centos.org/centos/8/AppStream/x86_64/os/

Note: There are cases where packages are made publicly available in the repositories without an associated CentOS advisory. In such cases, no Fixlets are provided by BigFix.

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 acquiring a Masthead file from a vendor or from HCL or 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*.

Patching method

BigFix offers more flexibility to the patch management solution by using native tools.

The Fixlets for all CentOS content use the Yellow dog Updater, Modified (YUM), the default patch manager for CentOS. 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.

Previously, the BigFix Patch for CentOS sites used a set of utilities that are called Endpoint Dependency Resolver (EDR) utilities to handle package dependencies on the endpoint. YUM replaces these EDR utilities and gives you more flexibility in terms of patch deployment and providing results that are in parallel with Red Hat and CentOS solutions. The following sites are available for

YUM utility configuration settings

The BigFix Patch for CentOS sites that apply the YUM utility use Fixlet settings in `/etc/yum.conf`. except for the following YUM configuration settings:

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

Identifying file relevance with Native tools content

The native tools captures file relevance in the same way as EDR. Both methods check for the relevance clause `exist lower version of a package, but not exist higher version of it`. If both tools are applied to the same deployment, the relevance results are the same.

Patching method matrix

The following table lists the applicable sites and features for each of the patching methods that are available for managing your CentOS endpoints.

Patching method	Applicable sites	Applicable features
Endpoint Dependency Resolution (EDR)	• Linux RPM Patching • Patches for CentOS 5* • Patches for CentOS 6*	• Download Plug-ins • RPM Deployment • Preference List
Native tools (YUM)	• Patching Support • Patches for CentOS 5 Native Tools • Patches for CentOS 6 Native Tools • Patches for CentOS7 Plugin R2	Download Plug-ins
Native tools (DNF)	Patches for CentOS 8	Download Plug-ins

*The **Patches for CentOS 5** site and **Patches for CentOS 6** site have been deprecated on December 31, 2016. BigFix Patch no longer provides content and support for these sites.

Chapter 2. Using the download plug-in

The download plug-ins, CentOS Plug-in and CentOS Plug-in R2, 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 4. CentOS Download Plug-in

Download Plug-in Name	Applicable Sites
CentOS Plug-in R2	Patches for CentOS6 Plugin R2 Patches for CentOS7 Plugin R2 Patches for CentOS 8
CentOS Plug-in	Patches for CentOS 5 Native Tools Patches for CentOS 6 Native Tools Patches for CentOS 7

Important: The **CentOS Plug-in**, **Patches for CentOS 6 Native Tools** site, and **Patches for CentOS 7** site are going to be deprecated in September 2017. You must switch to **CentOS Plug-in R2** of the said components at the soonest possible time to ensure continuous patching and uninterrupted access to the BigFix content.

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 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 CentOS Plug-in accesses the following URLs:

- `http://mirror.centos.org`
- `http://vault.centos.org`

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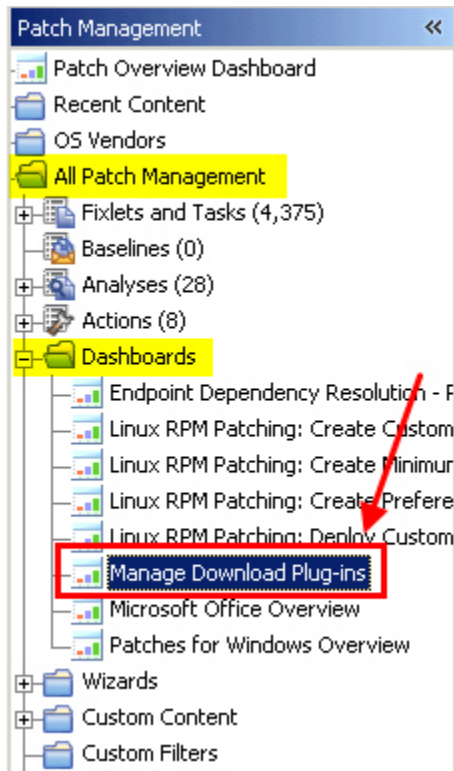
The dashboard displays all the servers and windows-only relays in your deployment. Select a server or relay to view all the plug-ins for that computer. The dashboard shows you also the version and status for each plug-in in one consolidated view.

Figure 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 CentOS Download Plug-in R2

Use the Manage Download Plug-ins dashboard to register the CentOS Download Plug-in R2 to install CentOS patches from the Patches for CentOS6 Plugin R2 and Patches for CentOS7 Plugin R2 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 **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 Download Plug-in R2.

Configuring the basic CentOS Download Plug-in R2 settings

Use the Manage Download Plug-ins dashboard to configure the proxy settings of the CentOS Plug-in R2.

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 CentOS Plug-in R2 configuration from the BigFix console. To configure the advanced settings for the CentOS Plug-in R2, you must edit the `plugin.ini` file. For details about how to do this, see Configuring the advanced CentOS Download Plug-in R2 settings *(on page 23)*.

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.
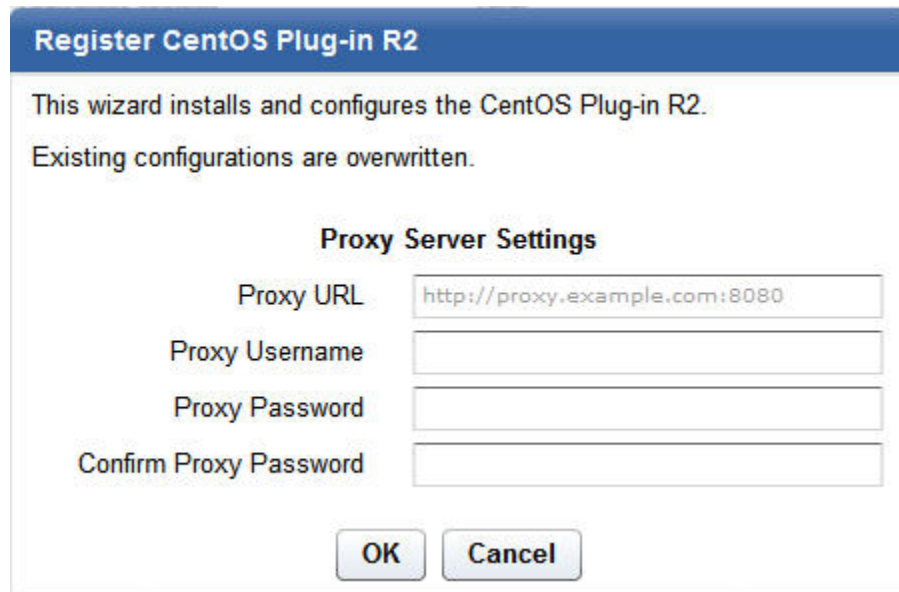
1. From the Patch Management domain, click **All Patch Management > Dashboards > Manage Download Plug-ins dashboard**.
2. From the Servers and Relays table, select the server on which the download plug-in is to be configured.
3. From the Plug-ins table, select **CentOS Plug-in R2**.
4. Click **Configure**.

   The Configure CentOS Plug-in R2 wizard displays.

Figure 5. Configure 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 configured the CentOS Download Plug-in R2.

Configuring the advanced CentOS Download Plug-in R2 settings

For advanced configurations, manually edit the CentOS Download Plug-in R2 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/CentOSR2Protocol`.

On Windows systems, the file is in the BigFix server installation directory. For example, `%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\CentOSR2Protocol`.

Figure 6. Example of the CentOS Download Plug-in R2 configuration file

```
[Logger]
file = logs/CentOSPluginR2.log
level = INFO


[UA]
proxy               =
proxyUser           =
proxyPass           =


primaryRepoListFile  = C:\Program Files (x86)\BigFix Enterprise\BES Server\
        GatherDBData\gather\Patching Support\CurrentSiteData\DLCentOSRepoList.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\CentOSR2Protocol` folder.

Setting the logging level

The logging level determines the amount of detail that is written to the `CentOSPluginR2.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/CentOSPluginR2.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 CentOS Download Plug-in R2 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 CentOS Download Plug-in R2 *(on page 26)*.

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 CentOS Download Cacher R2 for air-gapped environments *(on page 36)*.

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 37)*.

Without the sha1 download capability on an internet-enabled BigFix server
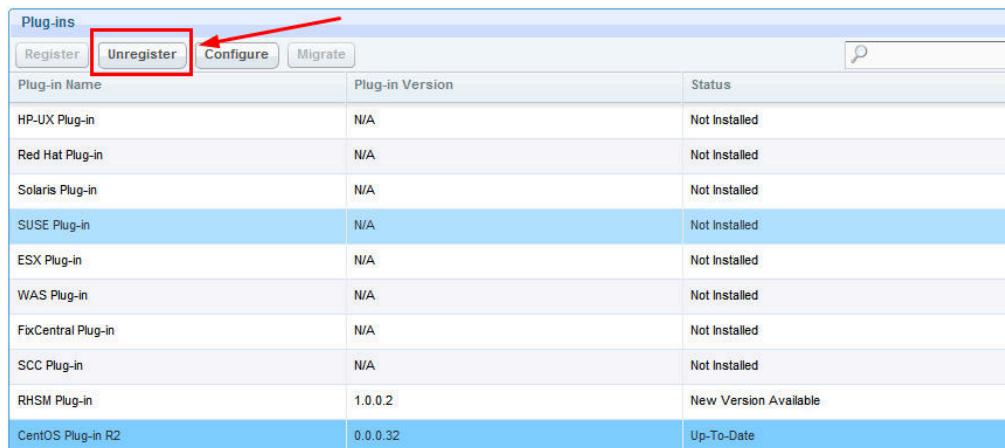
If for some reason you choose not to use the sha1 download capability to cache packages on the BigFix server's sha1 folder, you can use the local cache. For information, see Caching packages on the local cache folder *(on page 38)*.

Unregistering the CentOS Download Plug-in R2

Use the Manage Download Plug-ins dashboard to unregister the CentOS Download Plug-in R2.

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 **CentOS Plug-in R2**.
4. Click **Unregister**.

Figure 7. Unregister the CentOS download plug-in R2

Plug-in Name	Plug-in Version	Status
HP-UX Plug-in	N/A	Not Installed
Red Hat Plug-in	N/A	Not Installed
Solaris Plug-in	N/A	Not Installed
SUSE Plug-in	N/A	Not Installed
ESX Plug-in	N/A	Not Installed
WAS Plug-in	N/A	Not Installed
FixCentral Plug-in	N/A	Not Installed
SCC Plug-in	N/A	Not Installed
RHSM Plug-in	1.0.0.2	New Version Available
CentOS Plug-in R2	0.0.0.32	Up-To-Date

The Take Action dialog displays.

5. Select the target computer.

6. Click **OK**.

You successfully unregistered the CentOS download plug-in.

Upgrading the CentOS Download Plug-in R2

Use the Manage Download Plug-ins dashboard to upgrade the CentOS Download Plug-in R2 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 **CentOS Plug-in R2**.

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 CentOS Download Plug-in R2 installed.

Extending the CentOS Download Plug-in R2

You can configure the CentOS Download Plug-in R2 to download and cache packages from repositories that are not officially supported by BigFix. The CentOS Plug-in R2 can resolve package dependency for unsupported CentOS 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 CentOS subscription is active.
- Review the BigFix Patch for CentOS support information and verify the list of supported repositories:
   ◦ Supported platforms and updates *(on page 11)*
   ◦ Supported CentOS repositories *(on page 13)*

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

Step 1: Creating an extended repository list file

The CentOS Plug-in R2 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 CentOS Plug-in R2. Ensure that the Administrator or root user can read the file.
- It must contain the following format:

```
{CentOS Plug-in R2

   "<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 `DLCentOSRepoList.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 repositories with the BigFix CentOS Patching content, use the following OS keys as listed in the `DLCentOSRepoList.json` file:

- centos-8.1.1911-x64
- centos-8.0.1905-x64
- centos-8-x64
- centos-7.6.1810-x64
- centos-7.5.1804-x64
- centos-7.4.1708-x64
- centos-7.3.1611-x64
- centos-7.2.1511-x64
- centos-7.1.1503-x64

- centos-7.0.1406-x64
- centos-7-x64
- centos-6.8-x64
- centos-6.7-x64
- centos-6.6-x64
- centos-6.5-x64
- centos-6.4-x64
- centos-6.3-x64
- centos-6.2-x64
- centos-6.1-x64
- centos-6.0-x64
- centos-6-x64
- centos-6.8-x32
- centos-6.7-x32
- centos-6.6-x32
- centos-6.5-x32
- centos-6.4-x32
- centos-6.3-x32
- centos-6.2-x32
- centos-6.1-x32
- centos-6.0-x32
- centos-6-x32

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 `DLCentOSRepoList.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 CentOS mirror and vault repository lists:

```
http://mirror.centos.org/centos/<CentOS_version>/<repo_name>/<architecture>
```

```
http://vault.centos.org/<CentOS_version>/<repo_name>/<architecture>
```

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

```
{
 "centos-7-x64": [
  {"name": "EPEL 7", "baseurl": "https://dl.fedoraproject.org/pub/epel/7/x86_64"}
 ],
 "centos-6-x64": [
  {"name": "EPEL 6", "baseurl": "https://dl.fedoraproject.org/pub/epel/6/x86_64"}
 ]
}
```

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

Step 2: Updating the CentOS download plug-in R2 configuration file

Configure the CentOS Plug-in R2 to use an extended repository list file. The `plugin.ini` configuration file is overwritten when the CentOS Plug-in R2 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
   \CENTOSR2Protocol
   ```

   On Linux systems

   ```
   /var/opt/BESServer/DownloadPlugins/CENTOSR2Protocol
   ```

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 CentOS Plug-in R2 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 CentOS Plug-in R2 to only use the extended repository list, set the **onlyUseExtendedRepoListFile** field to yes.

   For example:

```
onlyUseExtendedRepoListFile= yes
```

If you want to set CentOS Plug-in R2 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 CentOS Download Cacher R2 to verify whether the extended repository has been set up correctly and that you can it. For details about the cacher, see CentOS Download Cacher R2 usage information *(on page 31)*.

To do this, run the `CentOSR2DownloadCacher.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: centos-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      :  CentOS_-_OS
3488      : 2018-02-28 17:23:36 : INFO      :  Success!


3488      : 2018-02-28 17:23:36 : INFO      :  Testing access to: centos-7-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      :  CentOS_-_OS
3488      : 2018-02-28 17:23:36 : INFO      :  Success!
```

Chapter 3. 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.

CentOS Download Cacher R2 usage information

Use the CentOS Download Cacher R2 to download and cache CentOS patches in air-gapped environments. This tool supports the **Patches for CentOS6 Plugin R2** and **Patches for CentOS7 Plugin R2** sites.

The CentOS Download Cacher R2 accesses the `http://mirror.centos.org` and `http://vault.centos.org` 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 CentOS Download Cacher R2 on a Windows system or a Linux system. For information about requirements, see BigFix 10.0 - 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 Plug-in 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 CentOS 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 CentOS version. This is because many packages are duplicated among

repositories with the same CentOS version (for example, centos-6.8-x64, centos-6.7-x64, centos-6.6-x64). Space is not saved if you only cache a single repository for each CentOS version (for example, centos-6.8-x64, centos-7.1-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 CentOS 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 CentOS 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 CentOS operating system version and architecture of a single CentOS repository.

The syntax to run this subcommand is:

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

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 centos-7-x64, centos-6-x32, centos-6.8-x32`

where:

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

Specifies the CentOS 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 centos-7-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 centos-7-x64
```

where:

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

Specifies the CentOS 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 centos-7-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 centos-7-x64 --pkg python-qrcode-core-5.0.1-1.el7.noarch.rpm
```

where:

--key *OS_key1,OS_key2,…*

Specifies the CentOS 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 python-qrcode-core-5.0.1-1.el7.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`
`liblcms1-1.17-77.12.1.x86_64.rpm,liblcms1-32bit-1.17-77.12.1.x86_64.rpm`.

Using the CentOS Download Cacher R2 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 CentOS 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 centos-6.8-x64, centos-6.7-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 "centos-6.8-x64" and "centos-6.7-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 CentOS 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 centos-7-x64, centos-6-x32
```

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 Plug-in 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 centos-7-x64, centos-6-x32
```

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 4. Using BigFix Patch for CentOS Linux™

Use the Fixlets on the Linux™ RPM Patching and the various Patches for CentOS Fixlet sites to apply patches to your deployment.

For information about the available Fixlet sites for CentOS Linux Enterprise, see Supported platforms and updates *(on page 11)*.

Patch content caching must be done through the download plug-in unless you are using an air-gapped environment or a custom repository. For more information, see the following topics:

- Using the download plug-in *(on page 16)*
- Using the download cacher *(on page 31)*

Patching using Fixlets

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

- Subscribe to the appropriate Fixlet sites.
- Activate the necessary analyses from the Patching Support site or Fixlet sites, such as the **Endpoint Dependency Resolution - Deployment Results** analysis to view the patch deployment results.
- Install and enable GPG keys on the endpoints. You can use the **Import RPM-GPG-KEY-centos-release** task to import the necessary keys.
- Set the appropriate configuration for the client `/var` directory.
    - If the `/var` directory is set to `noexec` option, specify a different directory to run the executable for patching. Otherwise, patching will fail. You can run the **Set the path for _BESClient_LinuxPatch_executable_directory** Fixlet and specify the alternative directory name.
    - If the `/var` directory has limited disk space, provide a directory with more space to cache the patch metadata. You can run the **Set the path for _BESClient_LinuxPatch_metadata_directory** Fixlet and specify the alternative directory name.

    Note: The specified directory path for both the `_BESClient_LinuxPatch_executable_directory` and `_BESClient_LinuxPatch_metadata_directory` settings must be a valid, absolute path name. It can contain only alphanumeric characters, forward slashes, and underscores.

- If you are using repositories, register the endpoints to the repository that contains the patches. You can use YUM or the CentOS Custom Repository Management dashboard to add a repository as a package source. For more information about the dashboard, see CentOS Custom Repository Management *(on page 51)*.
- If you are not using repositories, register the appropriate CentOS download plug-in. For more information about download plug-ins, see Using the download plug-in *(on page 16)*.

When running a CentOS Linux™ patch Fixlet®, you can also deploy a test run before applying the patch. You can view the Deployment Results analysis to determine if the dependencies have been successfully resolved and if an installation will be successful.

Note: When using the test run feature, the action reports back as Fixed, even if the test fails.

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 > CentOS**, and navigate to the patch content using the domain nodes.

Figure 8. Patch Management navigation tree
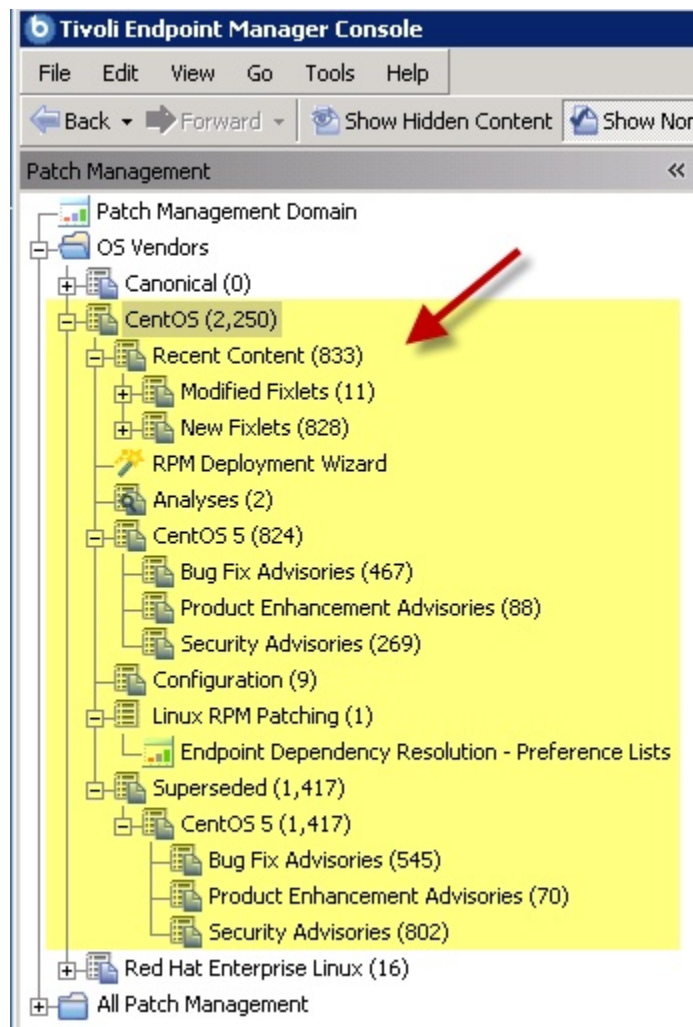
2. In the content that is displayed in the list panel, select the Fixlet that you want to deploy. The Fixlet opens in the work area.
3. Click the tabs at the top of the window to review details about the Fixlet.
4. Click **Take Action** to deploy the Fixlet.

> 📝 **Note:** If you are using a repository to be the package source, run the **Enable custom repository support - CentOS** task before taking any of the Fixlet action. The task configures the actions to use YUM to perform the necessary downloads from the configured repositories on the endpoints. When the custom repository support is enabled, the Fixlets stop downloading the metadata and packages through the Bigfix infrastructure and let YUM download the necessary files.

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

- You can start the 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.
- You can view the Red Hat bulletin for a particular Fixlet, select the **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**.

Use the RPM Deployment Wizard

Use the RPM Deployment Wizard to facilitate deployment of RPM packages.

> 📝 **Note:** RPM Deployment Wizard is removed since **Linux RPM Patching** site is deprecated. For more details, please refer to https://forum.bigfix.com/t/ibm-bigfix-patch-to-deprecate-and-end-support-for-linux-rpm-patching-site-on-september-29-2017/22405.

This wizard does not resolve dependencies and is only meant to handle simple deployment cases. In the Content Filters part of the navigation tree, click **OS Vendors > CentOS Linux > RPM Deployment Wizard**.

Figure 9. RPM Deployment Wizard in the navigation pane
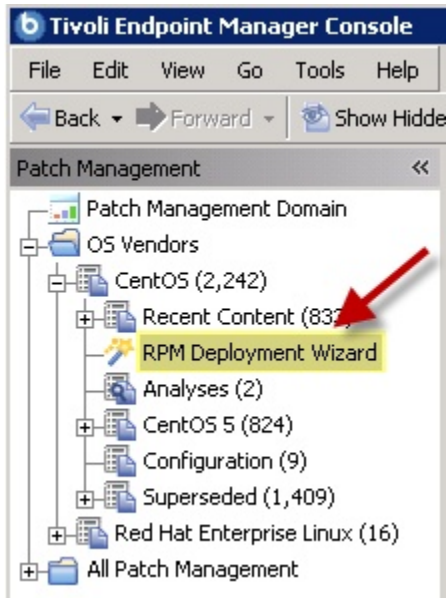
When the Wizard opens, enter the RPM package that you want to deploy.

Figure 10. Specify the source RPM package
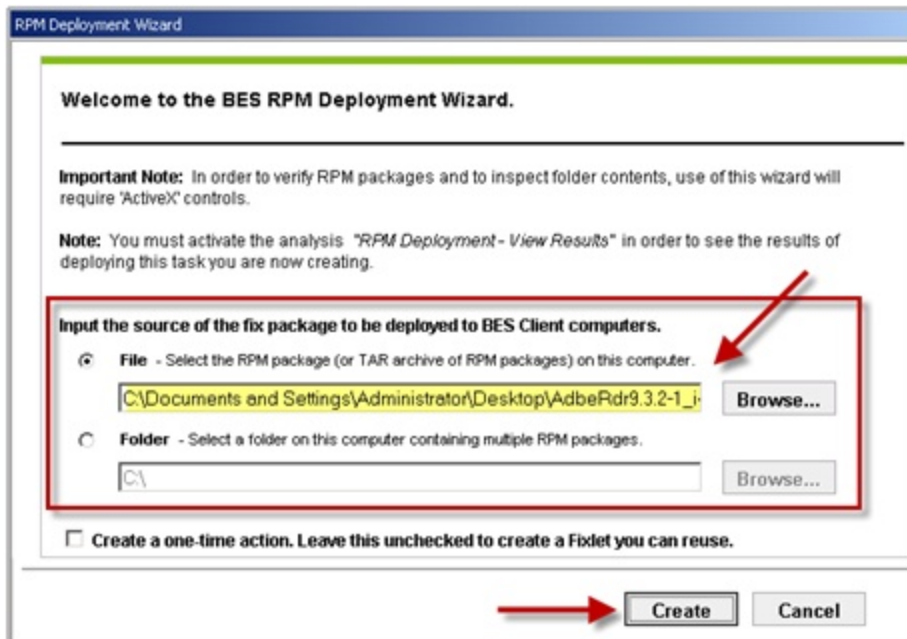
Select the check box if you want to create a one-time action, or leave blank to create a Fixlet® you can reuse. When complete, click **Create**. You see the following progress screen:

Figure 11. Progress screen - Creating archive
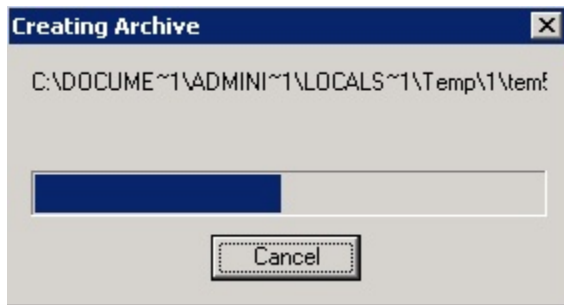
When the first Task window opens, click **OK** and enter your Private Key Password. Then click in the Actions box to start deployment.

Figure 12. Task window - Actions box
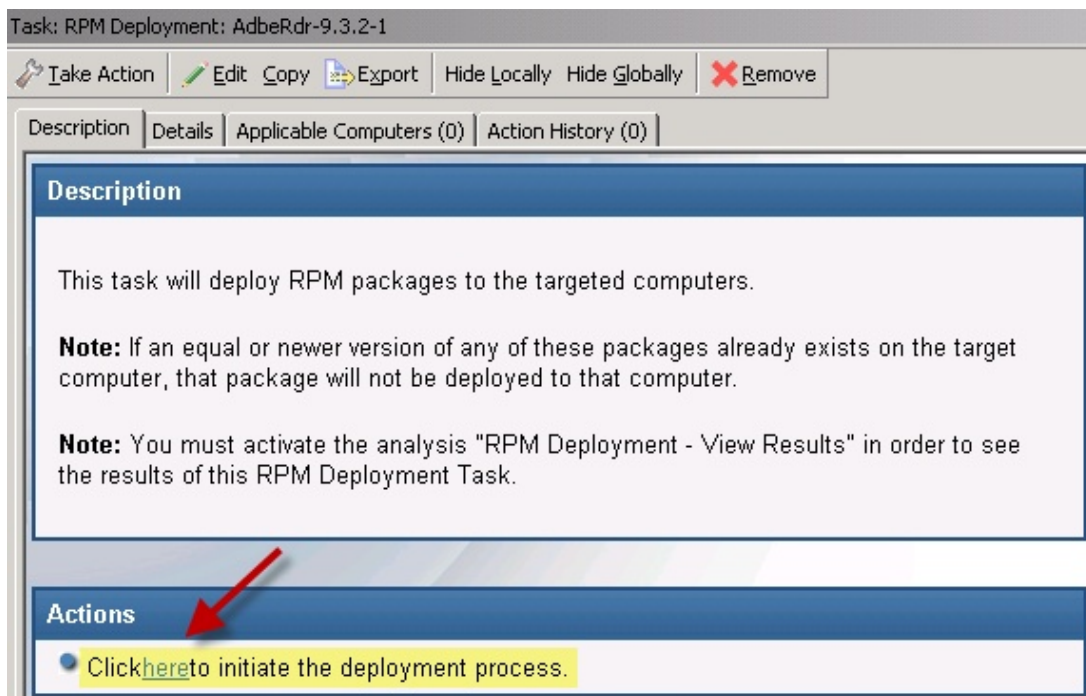
When the Take Action dialog opens, use the tabs across the top of the window to set parameters for this Action. Click **OK** and enter your Private Key Password. The Wizard deploys this action to your clients.

Supersedence

Please refer to Supersedence for Non-Windows to learn more about the concept of supersedence.

Patches for CentOS 8 site only supports BaseOS and AppStream repositories, For more information see Supported CentOS Repositories.

CentOS 8 repositories metadata now contains information related to latest available packages only, hence all our superseded fixlets are expected to fail with an error `No matching Package found`.

Once the fixlet is superseded, the fixlet is made non-relevant by default.

Client setting *value of setting "_BESClient_CentOS_EnableSupersededEval" of client* is removed for superseded fixlets and relevance for superseded fixlets will be set to **false***<Relevance>False</Relevance>.* You need to deploy only the latest available patches.

In order to deploy superseded fixlet using custom site, the relevant repository metadata and packages must be precedingly cached.

> **Note:** Best practice is to sync all the Baselines before proceeding with installation because having superseded fixlets in the Baseline would lead to errors and installation failures. Only latest available packages can be deployed with the baseline.

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 - CentOS 6** task, **Enable the Multiple-Package Baseline Installation feature - CentOS 7** task, or the **Enable the Multiple-Package Baseline Installation feature - CentOS 8** 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:**
>
> - Fixlets with packages that are stored in the CentOS Vault, which are indicated by the label `(vault x.y)` in the Fixlet title, cannot be deployed using the Multiple-Package Baseline Installation feature. You must deploy such Fixlets individually or in a regular baseline.
> - 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 CentOS 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 CentOS6 Plugin R2 site:

- Multiple-Package Baseline Installation - CentOS 6 - x86
- Multiple-Package Baseline Installation - CentOS 6 - x86_64

Available from the Patches for CentOS7 Plugin R2 site:

• Multiple-Package Baseline Installation - CentOS 7 - x86_64

Available from the Patches for CentOS8 site:

Multiple-Package Baseline Installation - CentOS 8 - x86_64

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 CentOS endpoint.

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

Delete CentOS 6 Package List File for Multiple-Package Baseline Installation

Deletes the package list file on targeted the CentOS Linux 6 computers.

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

Delete CentOS 7 Package List File for Multiple-Package Baseline Installation

Deletes the package list file on the targeted CentOS Linux 7 computers.

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

Delete CentOS 8 Package List File for Multiple-Package Baseline Installation

Deletes the package list file on the targeted CentOS Linux 8 computers.

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

For detailed information about using the multiple package installation feature, see Installing multiple packages in a baseline .

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 RPM-GPG-KEY-centos-release** 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 46)*.

📝 **Notes:**

- For CentOS 8, 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 CentOS.
    - 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 CentOS 6 Package List File for Multiple-Package Baseline Installation** task, **Delete CentOS 7 Package List File for Multiple-Package Baseline Installation** task, or the **Delete CentOS 8 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 13. Baseline component option
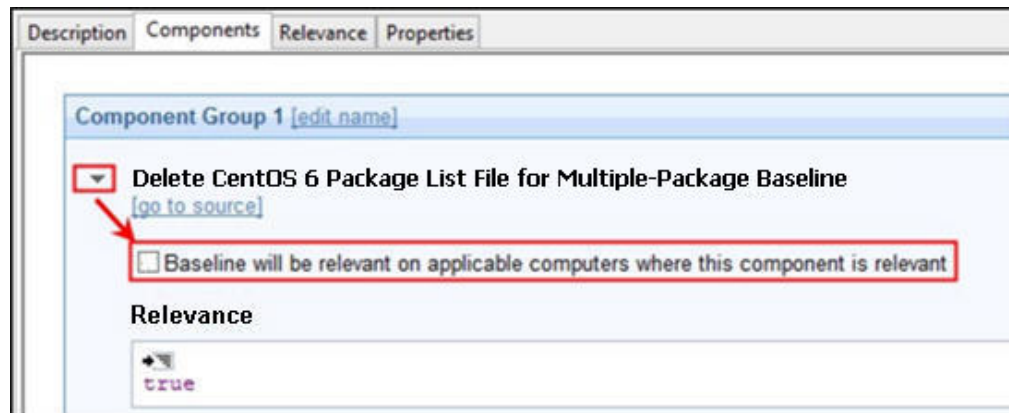
3. Add the appropriate **Enable the Multiple-Package Baseline Installation feature** task.
   Ensure that the **Baseline will be relevant on applicable computers where this component is relevant** option is not selected.
4. Selectively add the patch Fixlets in the baseline.
   Ensure that for all Fixlets 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: https://hclpnpsupport.service-now.com/csm?id=kb_article&sys_id=d288c2021b098c9477761fc58d4bcbdf

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. Custom repositories management

You can set up your custom repositories to manage patches for CentOS Linux version 5 and later. This solution allows for multiple repositories on the entire deployment.

With the custom repository support, the Fixlets in the CentOS native tools sites can use YUM to directly download packages from custom repositories instead of going through the package sources at *mirror.centos.org* and *vault.centos.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. For example, you can deploy custom software that you are hosting in your custom repositories, then use the **Install packages by using YUM** task from the **Patching Support** site to install software on the endpoints. For more information, see Installing packages from a custom repository *(on page 54)*.

Integrating your custom repository solution is made easy with the use of the **CentOS Custom Repository Management** dashboard. However, the dashboard does not add physical repositories; you must do this action separately. To learn about creating repositories, see https://wiki.centos.org/HowTos/CreateLocalRepos.

> ⚠️ **Important:** You must register the CentOS 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.

CentOS Custom Repository Management dashboard

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

The CentOS Custom Repository Management dashboard allows Fixlets to use YUM for downloads instead of using the standard BigFix downloading infrastructure. The dashboard also allows you to register repositories to use the YUM 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 > CentOS Custom Repository Management**. Activate the **Repository Configuration - CentOS** 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 CentOS 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 CentOS Custom Repository Management dashboard does not support the creation of a physical repository server. You must create the repository separately. For more information about creating repositories, see the following resource: https://wiki.centos.org/HowTos/CreateLocalRepos

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

Adding a repository

Use the CentOS 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 - CentOS** analysis to populate the dashboard with endpoint and repository information.

1. From the **CentOS 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 CentOS endpoints to a repository *(on page 52)*.

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 54)*.

Registering CentOS endpoints to a repository

You can use the CentOS 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 - CentOS** analysis if you have not yet done so.

1. From the **CentOS 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 YUM 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 - CentOS** task to use the repository during patch deployment.

Unregistering CentOS endpoints from a repository

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

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

1. From the **CentOS 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 **CentOS 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 CentOS 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 - CentOS** 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 **CentOS 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.

Checking for YUM package updates

Use the **YUM: Check Available Package Updates** task to identify and determine whether your CentOS Linux endpoints contain applicable YUM package updates for installation.

The **YUM: Check Available Package Updates** task uses the `yum check-update` command to determine which updates are available for the installed packages on your endpoints. This task is available from the **Patching Support** site.

Before you run the task, ensure that the **YUM Logs** analysis (ID #28) is activated. The result details of the YUM package updates that are applicable to your endpoints are listed in a new column in that analysis called **YUM check-update output**.

The task uses the repository in your deployment, so ensure that the YUM packages are available in your repository when you run the task.

Installing packages from a custom repository

BigFix provides a task to easily install and update packages on CentOS Linux version 5 and later endpoints that are registered to custom repositories.

- Subscribe to the **Patching Support** site to access the installation task named as **Install packages by using YUM**.
- Configure a custom repository from the CentOS Custom Repository Management dashboard. For more information, see CentOS Custom Repository Management dashboard *(on page 51)*.
- Register the CentOS R2 download plug-in through the Manage Download Plug-ins dashboard. The data generated by the download plug-in may not be used, but it is crucial that you register the plug-in to avoid download errors.
- Ensure that the configured repository is up-to-date and contains the required packages and metadata.

Use the **Install packages by using YUM** task to install or update the packages on endpoints.

You can use the package name or Common Vulnerabilities and Exposures (CVE) ID number to specify the selected packages for installation.

You can also update all the installed packages on the endpoint with later available versions that are in your custom repository.

The YUM commands for each of the available actions are as follows:

yum install *<package_name1> <package_name2>*

Updates or installs a package with a specific name. Multiple package update or installation is acceptable. Use a space to separate the package names.

yum update

Updates all the installed packages on the endpoint.

yum patch --cve=*<cve_number>*

Updates a package with a specific CVE ID number. The task fails if no CVE ID number is specified and it only accepts a single CVE reference.

yum patch --date=*<YYYY-MM-DD>*

Installs all patches that were issued until the specified date.

Command options are supported as extra flags for the `yum install` and `yum update` commands only. For detailed usage information, see the yum man page.

This task also provides actions to test the packages for installation, without installing the packages on the endpoints.

1. From the Patch Management domain, click **All Patch Management > Fixlets and Tasks**.
2. Select the **Install packages by using YUM** task to install custom packages on endpoints.
3. In the Task pane, review the description and follow the instructions in the Actions box to deploy an action.
4. Depending on the action that you selected, provide the necessary information and click **OK**.

> **Note:** To update all installed packages on the endpoint, select the action to install packages, but do not specify any package name.

5. In the Take Action pane, select the endpoints on which the packages are to be installed or updated.
6. Click **OK**.

Chapter 7. YUM transaction management

View YUM transaction history and manage transactions through the **YUM Transaction History** dashboard, which works with CentOS Linux version 5, 6 and 7.

Note: YUM transaction history dashboard is not supported in CentOS 8.

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 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 **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.

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

Appendix B. Troubleshooting

When problems occur when patching CentOS endpoints, review the log files to determine what went wrong and how to correct the errors.

Log files

Enhanced logging with clearer error reporting and error handling to improve troubleshooting.

CentOSPluginR2.log

Lists the results of the downloads related to the execution of the CentOS Download Plug-in R2. The amount of information depends on the logging level.

The log can be found in the following locations:

- On Windows systems: `%PROGRAM FILES%\BigFix Enterprise\BES Server \DownloadPlugins\CentOSR2Protocol`
- On Linux systems: `/var/opt/BESServer/DownloadPlugins/CentOSR2Protocol`

The following log files can be found in the client folder in the directory `/var/opt/BESClient/EDRDeployData`.

EDR_DeploymentResults.txt

Lists the results of the EDR deployment and the YUM output.

register-repo.log

Lists the results of the repository registration action of the CentOS Custom Repository Management dashboard.

unregister-repo.log

Lists the results of the unregister repository action of the CentOS Custom Repository Management dashboard.

yum_history.log

Lists the results from the YUM Transaction History dashboard.

Other useful log files:

yum.log

This is the official log that YUM generates by default in `/var/log/yum.log`. It lists all the YUM-related operations and transactions.

Download plug-in logging levels

The logging level determines the amount of detail that the CentOS Download Plug-in R2 writes to the log files. Set the logging level in the `%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins \CentOSR2Protocol\plugin.ini` file.

Note: Logging level values are case-sensitive.

The following logging levels are listed in order of increasing amount of information logged:

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.

Note: Setting the logging level to DEBUG increases the amount of information to log, which might have an impact on performance. You must only increase the logging level to DEBUG when investigating an issue.

Enable the client debug log

Use the **Configure the client setting to enable the Debug Log for Linux Patching** Fixlet (ID #57) from the Patching Support site.

Prefetch plug-in error

If you have taken an action on a Fixlet that failed on a line with `execute prefetch plug-in` in the Actionscript. Subsequent calls to the same prefetch plug-in from any Actionscript might fail on the endpoint. The script might have been blacklisted, causing the prefetch plug-in error.

To verify, check the client log. You will see either of the following messages for the Fixlet action executing the prefetch plug-in:

```
execute prefetch plug-in' didn't complete within 300 seconds. Black listing plug-ins

matching the sha1 hash of 'name of 'bash' until agent is restarted.
```

```
Execute prefetch plug-in attempting to reuse plug-in which took too long earlier.
```

To resolve this issue, do the following actions:

1. Restart the BigFix client to clear the blacklist.
2. Set the `_BESClient_ActionManager_PrefetchPlugInTimeoutSeconds` client configuration setting with sufficient time for the patch to install and resolve dependencies. This client setting indicates how long the client should wait

before blacklisting the script. You can use the **Change Timeout for Prefetch Plugins** task, available from the Patching Support site, to set the setting to 30 minutes (1800 seconds).

> 📝 **Note:** The `_BESClient_ActionManager_PrefetchPlugInTimeoutSeconds` setting varies based on the endpoint and the Fixlet being installed. To get the desired value, take the slowest endpoint and increase the setting to a high number, such as 3,000 seconds, then run a large Fixlet and see how long it takes. You can then take that number and multiple it by two. Alternatively, set the client setting to 600 seconds and adjust it accordingly if the suggested value does not work for you.

Error when /var is mounted as noexec

All available Fixlets use an executable that by default runs directly from the `/var` directory, a partition on the endpoint. The Fixlets will not work when `/var` is set with the `noexec` option, regardless of whether the CentOS Download Plug-in R2 or Custom Repository solution is used. Therefore, ensure that the `/var` directory is not set with the `noexec` option by doing the following steps:

1. Check the client log to see if the prefetch plug-in returned the `exit code 126`. For example:
2. Run `mount` as the root user to check the mount option that is currently used:

```
[root@host ~]# mount

/dev/mapper/vg_data-lv_root on / type ext4 (rw)

proc on /proc type proc (rw)

sysfs on /sys type sysfs (rw)

devpts on /dev/pts type devpts (rw,gid=5,mode=620)

tmpfs on /dev/shm type tmpfs (rw)

/dev/sda1 on /boot type ext4 (rw,nodev)

/dev/mapper/vg_data-lv_var on /var type ext4 (rw,noexec,nosuid,nodev)

none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
```

If `/var` is set to `noexec`, you must take one of the following actions:

- Remove the `noexec` mount option.
- Move `/var/opt/BESClient` to a different partition, which is not `noexec`, and create a symbolic link to it in its place.
- Run the **Set the path for _BESClient_LinuxPatch_executable_directory** Fixlet and specify an alternative directory to run the executable for patching. The directory path must be a valid, absolute path name. It can contain only alphanumeric characters, forward slashes, and underscores.

Missing GPG key

Deploy the **Import RPM-GPG-KEY-centos-release** task (ID# 301), available in the Fixlet site, before deploying any of the patch Fixlets to avoid issues during deployment.

Repository metadata is too large

The repository metadata that is used in patching is provided by the vendor and can be large. If the `/var` directory does not have enough space to store the metadata, set an alternative directory with sufficient space to store the metadata on the endpoint. You can use the **Set the path for _BESClient_LinuxPatch_metadata_directory** to set the directory where the repository metadata will be created.

Null error when configuring BigFix Patch download plug-ins

When the BigFix server and the BigFix client on the BigFix server do not have the same version, users might encounter a null error. The error occurs because BigFix server 8.x and 9.x versions handle encryption differently. The version of the client on the BigFix server is used to determine the BigFix server version and it is assumed that the version is the same for the BigFix server and the client on the BigFix server.

Ensure that the version of the BigFix server and BigFix client on the BigFix server match to avoid null errors when configuring the download plug-in. At a minimum, the version must be on the same major version level, for example 8.x or 9.x.

Appendix C. Frequently Asked Questions

To better understand BigFix Patch for CentOS, read the following questions and answers.

The Manage Download Plug-ins dashboard is not reflecting any data. What do I do?

Here are some steps you can do to troubleshoot the issue:

- Gather the latest **Patching Support** site.
- Activate the **Download Plug-in Versions** analysis, available from the **Patching Support** site.
- Clear the BigFix console cache.

What are superseded patches?

Superseded Fixlets are Fixlets that contain outdated packages. If a Fixlet® is superseded, then there exists a newer Fixlet® with newer versions of the packages. The newer Fixlet® ID can be found in the description of the superseded Fixlet®.

Where are the deployment logs located on endpoints?

The logs are located in a folder called `EDRDeployData` in the client folder at `/var/opt/BESClient/EDRDeployData`.

Why is my action reporting back as a failed download?

Make sure your download plug-in has been updated to the latest version and is registered with the correct credentials.

What do I do when an action reports back with an installation failure?

Check to see if the conflict is caused by a vendor-acquired package. These must be removed for the installation to occur.

I am not able to patch an Fixlets on endpoints with the `/var` directory mounted as `noexec`. What do I do?

For the workaround, see Error when /var is mounted as noexec *(on page 64)*.

How do I verify if the download plug-in was registered correctly?

Run a Fixlet with an action task to verify if the download plug-in is registered correctly. Verify that the patch download is successful. Otherwise, you might need to unregister the download plug-in and register it again.

How do I register a download plug-in? Do I use the register download plug-in task or the Manage Download Plug-in dashboard?

To register a download plug-in, you must use the Manage Download Plug-in dashboard in the Patching Support site. Existing register download plug-in tasks are being deprecated. To learn more about plug-in registration, see Registering the CentOS Download Plug-in R2 *(on page 19)*.

> **Note:** You must also use the Manage Download Plug-in dashboard to unregister, configure, and upgrade download plug-ins. The existing unregister and edit download plug-in tasks are being

deprecated. For more information about the dashboard, see the topic on Manage Download Plug-ins dashboard in the BigFix Knowledge Center.

Which version of the YUM native tools must be used?

The Patches for CentOS native tools site requires version 3.2.19-18 or later.

An action failed and the logs contain YUM-specific errors. How do I troubleshoot the failed action?

For more information about YUM and errors that are related to it, see the YUM documentation at http://yum.baseurl.org and the YUM-related articles in the Red Hat Customer Portal.

What version of YUM is required to use the CentOS Custom Repository dashboard?

You must have a minimum YUM version of 3.2.19-18.

Which CentOS Linux is supported by the CentOS custom repository dashboard?

The dashboard supports CentOS Linux versions 5, 6, and 7.

Which version of the BigFix server supports the CentOS Custom Repository dashboard?

The CentOS Custom Repository dashboard supports version 9.0 and later of BigFix.

When deploying patches, should I use the existing method or go through the custom repository? Can the two methods co-exist?

The two methods can exist together. However, when deploying patches for single clients, you must choose between using the native tools or through the custom repository method.

How are dependencies resolved the CentOS Custom Repository dashboard is used?

YUM uses the metadata to resolve the dependencies to know which packages are needed.

Can the custom repositories be used for software installation?

Yes, you use custom repositories for software installation. To use custom repositories for software installation, follow these steps:

1. Ensure that the clients are registered through the Custom Repository dashboard.
2. Create a Fixlet in the custom site with the action`yum install<space><package name>`. Ensure that you set the correct relevance or success criteria, that is, whether the Fixlet takes action against that client or endpoint. To learn more about creating Fixlets, see the *Console Operator's Guide*.

Note: It is important to ensure that the repositories are updated. Actions can fail if the packages are not available.

Can previously configured repositories be configured again?

Yes, you can configure again a previously configured repository.

From the logs, can I tell if I am using the normal YUM process to the repository in the log?

Yes, the log indicates whether the normal YUM process in the repository is used.

What is the difference between registering a repository and importing a repository?

Use the import feature if you have existing repositories that are not included in the Repositories list in the dashboard. Use the register feature if you already have a repository in the Repository list, but you still need to link the repository with the endpoint.

What happens when the repository does not contain the package?

When a package is not found, the Fixlet fails. You can troubleshoot from `EDR_DeploymentResult.txt`, which is where the YUM output is logged.

What happens if there are issues with the custom repository solution?

Users that encounter issues with the custom repository resolution can revert to the standard BigFix server solution. Users can run the task that is called **Disable custom repository support - CentOS**.

In the Endpoints tab of the CentOS Custom Repository Management dashboard, are the repositories that are listed in the lower part of the window that is used in sequence?

There is no sequence in the repositories that are listed in the **Endpoints** tab. When YUM queries the repositories, the repository that first gets the fetch query replies, including the package and its dependencies.

Through the CentOS Custom Repository Management dashboard, a patch was deployed through a custom repository that is not a mirror of the manufacturer site. The deployment failed and the EDR logs indicate that the rpm files could not be opened. What should I do?

When a custom repository that is not a mirror of the vendor site is used, it is possible that the default gpgcheck is being done as part of the installation and the gpg signature files might not be included. The rpm files might not be checked for authenticity and the installation might fail. To resolve this, ensure that when you register the endpoints in the CentOS Custom Repository Management dashboard, you added 'gpgcheck=0' to Additional fields.

Can I install several packages using 'Task: Install packages by using YUM'?

Yes, you can install several packages with the task. Use a space to separate the rpm names.

What version of YUM does the YUM Transaction History dashboard support?

You must have a minimum YUM version of 3.2.28.

Which version of CentOS Linux does the YUM Transaction History dashboard support?

The dashboard supports CentOS Linux versions 5, 6, and 7.

In the YUM Transaction History dashboard, what is the difference between rollback and undo?

The rollback command will undo all transactions up to the point of the specified transaction. The undo command will revert the selected transaction only.

What is the difference between the YUM Transaction History log and YUM Logs analyses?

Patch Management for CentOS Linux generates the YUM Transaction History log, which records the results of the actions that are taken in the YUM Transaction History dashboard. The log is located in `/var/opt/BESClient/EDRDeployData/yum_history.log`.

The YUM log is the official log that is generated by YUM in `/var/log/yum.log` by default. To change the default location, modify the log file settings in `/etc/yum.conf`.

Does the YUM Transaction History dashboard show only `yum -update all`?

Aside from `yum -update all`, the **Command Line** column in the dashboard also shows the different transactions such as the following installation commands.

- `install bzip2`
- `install net-tools`
- `install vim enhance`
- `install wget`

To learn more about YUM commands, see the Red Hat Enterprise Linux website: https://access.redhat.com/.

How does the YUM Transaction History dashboard work if you have configured the BigFix client to use YUM repository?

The YUM Transaction History dashboard does not adversely affect your deployment if you have configured the BigFix client to use YUM repositories. If you already use a local YUM repository, it might be easier to provide the packages for rollback.

The client logs contains a prefetch plug-in error that prevents the Fixlet from completing successfully. What is causing the error? What should I do?

The ActionScript that was running on the endpoint might have been blacklisted, causing the prefetch plug-in issue.

To resolve this issue, restart the BigFix client to clear the blacklist. To prevent the script from being blacklisted, set the `_BESClient_ActionManager_PrefetchPlugInTimeoutSeconds` client configuration setting with sufficient time to process the patch. For more information, see Prefetch plug-in error *(on page 63)*.

Can I set the log level for the CentOS Download Plug-in R2?

You can set the download plug-in to generate log messages depending on level of information that you need.

The logging level determines the amount of detail that the CentOS Plug-in R2 writes to the log files. Set the logging level in the `%PROGRAM FILES%\BigFix Enterprise\BES Server \DownloadPlugins\CentOSR2Protocol\plugin.ini` file.

> **Note:** Logging level values are case-sensitive.

The following logging levels are listed in order of increasing amount of information logged:

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.

Note: Setting the logging level to DEBUG increases the amount of information to log, which might have an impact on performance. You must only increase the logging level to DEBUG when investigating an issue.

Where can I find the log for the CentOS Download Plug-in R2? What are the possible log levels

Logging is controlled by the `plugin.ini` file. It is located with the download plug-in executable. By default, it is located in `%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\CentOSR2Protocol` on Windows systems. On Linux system, it is in `/var/opt/BESServer/DownloadPlugins/CentOSR2Protocol`. The log file is rotated on a daily basis, which means that a new log file is created and the old log file is renamed with the date that it is created from.

Will the CentOS Download Plug-in R2 configuration file (`plugin.ini`) be overwritten when there is a newer version of the download plug-in?

No, the configuration file will not be overwritten. The only time the configuration file is overwritten is when the download plug-in is reconfigured.

Can I reconfigure the CentOS Download Plug-in R2 proxy after registration?

Yes, you can update the proxy settings by configuring the download plug-in from the Manage Download Plug-ins dashboard.

What happens if I edit the `DLCentOSRepoList.json` file with more repositories?

The changes that you make will be deleted when BigFix refreshes the Patching Support site as the file will be overwritten.

I have a subscription for extension products, can I configure the CentOS Download Plug-in R2 to access their assigned repositories?

Yes, you can. For more information, see Extending the CentOS Download Plug-in R2 .

Where should I save the extended repository list file?

The file can be stored in any location the download plug-in has access to. You must ensure that the BigFix Server has permissions to access the location.

I am not able to install any packages since I upgraded to the CentOS Plug-in R2. All tasks result in the following line: Failed add prefetch item {concatenation " ; " of lines of file (parameter "EDR_PkgRequest")}. What is wrong?

The BigFix Enhanced Security option *-requireSHA256Downloads* or **Require SHA-256 Downloads** option in the BigFix Administration tool might be enabled. This option configures all download verification to use only the SHA-256 algorithm. The CentOS Download Plug-in R2 might fail due to certain CentOS repository metadata, which do not contain SHA-256 values for the 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.

Can I configure the CentOS Download Plug-in R2 to only use the extended repository list?

Yes, you can by setting the `onlyUseExtendedRepoListFile` flag in the `plugin.ini` to `yes`.

When deploying multiple Fixlets in a baseline, is it possible to skip the broken dependencies and continue with the installation for the rest of the packages?

The **Multiple-Package Baseline Installation** task skips packages with broken dependencies whenever possible. Packages with dependency issues with CentOS cannot be skipped. Another scenario where packages are not skipped is when dependency errors occur during installation, as indicated by the following error message: `File conflicts happen when two packages attempt to install files with the same name but different contents`. In such cases, the installation is canceled and no patches will be installed on the endpoints.

What are the possible causes of failure using the multiple-package baseline installation method?

The Fixlets might have failed to install due to the following reasons:

- A custom site contained multiple baselines with the multiple-package baseline installation task running at the same time.
- Two or more Fixlets required an update to multiple versions of the same package.
- Two or more Fixlets required an update to the same package dependencies.
- A Fixlet was immediately deployed after a baseline ran the multiple-package installation method. Not enough time was allowed to complete all YUM transactions and refresh the status on the endpoints for the multiple-package installation.
- The YUM tool have not been patched, but a CentOS release package is included in the baseline.
- The **Enable the Multiple-Package Baseline Installation feature** task must be in the same baseline as the rest of the content. It should be added before the patch Fixlets and multiple-package installation tasks.

- The Multiple-Package Baseline Installation feature only works when both the enable and installation tasks exist in the same baseline. For more information, see Installing multiple packages in a baseline *(on page 47)*.
- The **Enable the Multiple-Package Baseline Installation feature** task must be added after any of these cleanup tasks: **Delete CentOS 6 Package List File for Multiple-Package Baseline Installation** or **TROUBLESHOOTING: CentOS 7 Patching Deployment Logs - Cleanup**.
- The installation task with the correct CentOS distribution, operating system version, service pack level, and architecture must be added at the end of the baseline.

My endpoints are on air-gapped environment, how should I configure BigFix to patch these endpoints?

For air-gapped environments, ensure to mirror the supported CentOS repositories that host the packages needed to patch the endpoints. To do this, use the CentOS Download Cacher R2 to build the local repository on a location which the BigFix server has access to. This location is known as the local cache.

> **Note:** The local cache must contain all the required repositories to avoid dependency resolution issues.

Configure the download plug-in configuration file, `plugin.ini`, to use the local cache when downloading files during deployment. Follow these steps:

1. Set the local cache configuration, `localCache`, to the location of the files that were downloaded using the download cacher tool.

   > **Note:** This location must be accessible to the BigFix server.

2. Set the `localCacheOnly` flag to `yes`. to download files from the download cache only and not from the vendor site.

I ran a baseline with the Multiple-Package Baseline Installation task. How can I view the list of Fixlets that failed in that baseline?

You can monitor of the overall progression of the deployment of the baseline and view the status of each sub-action in detail by using the **View Action Info** dialog.

To access this dialog:

1. Click the **Action** icon in the navigation tree.
2. Select an action in the Actions List Panel.
3. Select the **Computers** tab in the Work Area.
4. Right-click any computer in the list.
5. Either select **Show Action Info** from the context menu or select **Show Action Info** from the **Edit** menu.

For more details about the failed Fixlets, check the client log on the target endpoints at `/var/opt/BESClient/__BESData/__Global/Logs`.

The baseline that I ran with the multiple-package installation task completed successfully, but why does it still show as relevant.

It might be due to the Fixlet components that failed to install the packages with broken dependencies. The **Multiple-Package Baseline Installation** task, by default, ignores broken dependencies to allow packages without dependency issues to successfully be applied on the target endpoint.

Is there a way to check if the supported CentOS repositories can be accessed?

From the command line, you can use the `--check-baserepos` and the `--check-allrepos` commands to check if the supported CentOS repositories can be accessed. For more information about these commands, see CentOS Download Cacher R2 usage information *(on page 31)*.

For a list of the CentOS repositories that BigFix supports, see the Support Matrix.

Is there a way to check for the required space when caching repository metadata and packages through the CentOS Download Cacher R2 tool?

Use the `check-storagereq` subcommand to check for the storage space requirement.

How can I save space when caching repository metadata and packages through the CentOS Download Cacher R2 tool?

Space-saving benchmarks are established with the use of the `--sha1_download_dir`.

Using the `--sha1_download_dir` have shown significant decrease in storage size, download size, and time when caching multiple repositories of the same CentOS version. This is because many packages are duplicated among repositories with the same CentOS version (for example, centos-6.8-x64, centos-6.7-x64, centos-6.6-x64). Space is not saved if you only cache a single repository for each CentOS version (for example, centos-6.8-x64, centos-7.1-x64).

An example output is as follows:

```
Caching centos-6.8-x64, centos-6.7-x64, centos-6.6-x64
(with --sha1_download_dir):
Total Repo Metadata and Packages will take up 28 GB of space
instead of 37 GB (23% space saved)
```

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.

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.