Red Hat JBoss Web Server 6.0

Installation Guide

Install and Configure Red Hat JBoss Web Server 6.0

Last Updated: 2024-03-15

Red Hat JBoss Web Server 6.0 Installation Guide

Install and Configure Red Hat JBoss Web Server 6.0

Legal Notice

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at
http://creativecommons.org/licenses/by-sa/3.0/
. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

Install, upgrade, and perform basic configuration of Red Hat JBoss Web Server on supported operating systems.

Table of Contents

PROVIDING FEEDBACK ON RED HAT JBOSS WEB SERVER DOCUMENTATION 4

MAKING OPEN SOURCE MORE INCLUSIVE 5

CHAPTER 1. INTRODUCTION TO RED HAT JBOSS WEB SERVER INSTALLATION 6
1.1. JBOSS WEB SERVER COMPONENTS 6
1.2. DIFFERENCES BETWEEN THE APACHE TOMCAT DISTRIBUTIONS THAT RED HAT PROVIDES 6
Apache Tomcat versions 7
Differences between JBoss Web Server and RHEL distributions of Apache Tomcat 7
Differences between the JBoss Web Server and RHEL documentation sets 10
1.3. JBOSS WEB SERVER OPERATING SYSTEMS AND CONFIGURATIONS 10
1.4. JBOSS WEB SERVER INSTALLATION METHODS 10
1.5. JBOSS WEB SERVER COMPONENT DOCUMENTATION BUNDLE 11

CHAPTER 2. INSTALLING JBOSS WEB SERVER ON RED HAT ENTERPRISE LINUX FROM ARCHIVE FILES 12
2.1. PREREQUISITES 12
2.1.1. Installing a JDK by using the DNF package manager 12
2.1.2. Installing a JDK from a compressed archive 13
2.1.3. Red Hat Enterprise Linux package requirements 13
2.2. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR A BASE RELEASE OF JBOSS WEB SERVER 14
2.3. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR JBOSS WEB SERVER PATCH UPDATES 14
2.4. MANAGING JBOSS WEB SERVER BY USING SYSTEMD WHEN INSTALLED FROM AN ARCHIVE FILE 15
2.5. JBOSS WEB SERVER CONFIGURATION FOR MANAGING ARCHIVE INSTALLATIONS FROM THE COMMAND LINE 17
2.5.1. Setting the JAVA_HOME environment variable for Apache Tomcat 17
2.5.2. Creating a Tomcat user and group 17
2.5.3. Granting the Tomcat user access to JBoss Web Server 18
2.6. STARTING JBOSS WEB SERVER FROM THE COMMAND LINE WHEN INSTALLED FROM AN ARCHIVE FILE 19
2.7. STOPPING JBOSS WEB SERVER FROM THE COMMAND LINE WHEN INSTALLED FROM AN ARCHIVE FILE 19
2.8. SELINUX POLICIES FOR JBOSS WEB SERVER 19
2.8.1. SELinux policy information for jws6-tomcat 19
2.8.2. Installing SELinux policies for a JBoss Web Server archive installation 20
2.9. CHANGING THE UID AND GID FOR THE TOMCAT USER AND GROUP 21

CHAPTER 3. INSTALLING JBOSS WEB SERVER ON RED HAT ENTERPRISE LINUX FROM RPM PACKAGES . 24
3.1. PREREQUISITES 24
3.1.1. Installing a JDK by using the DNF package manager 24
3.1.2. Installing a JDK from a compressed archive 25
3.1.3. Red Hat Enterprise Linux package requirements 25
3.2. ATTACHING SUBSCRIPTIONS TO RED HAT ENTERPRISE LINUX 26
3.3. INSTALLING JBOSS WEB SERVER FROM RPM PACKAGES BY USING DNF 26
3.4. STARTING JBOSS WEB SERVER WHEN INSTALLED FROM RPMS 27
3.5. STOPPING JBOSS WEB SERVER WHEN INSTALLED FROM RPMS 28
3.6. CONFIGURING JBOSS WEB SERVER SERVICES TO START AT SYSTEM STARTUP 28
3.7. SELINUX POLICIES FOR JBOSS WEB SERVER 28
3.7.1. SELinux policy information for jws6-tomcat 28
3.7.2. Enabling SELinux policies for a JBoss Web Server RPM installation 29

CHAPTER 4. INSTALLING JBOSS WEB SERVER ON MICROSOFT WINDOWS 30
4.1. INSTALLING A JDK ON MICROSOFT WINDOWS 30

4.2. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR A BASE RELEASE OF JBOSS WEB SERVER 30
4.3. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR JBOSS WEB SERVER PATCH UPDATES 31
4.4. JBOSS WEB SERVER CONFIGURATION ON MICROSOFT WINDOWS 32
4.4.1. Setting environment variables for JBoss Web Server on Microsoft Windows 32
4.4.2. Installing the Tomcat service on Microsoft Windows 32
4.4.3. Configuring folder permissions for JBoss Web Server services on Microsoft Windows 33
4.5. STARTING JBOSS WEB SERVER ON MICROSOFT WINDOWS 33
4.6. STOPPING JBOSS WEB SERVER ON MICROSOFT WINDOWS 34

CHAPTER 5. ENABLING HTTP/2 FOR THE RED HAT JBOSS WEB SERVER 36
5.1. PREREQUISITES 36
5.2. ENABLING HTTP/2 FOR A CONNECTOR 36
5.3. VIEWING JBOSS WEB SERVER LOGS TO VERIFY THAT HTTP/2 IS ENABLED 37
5.4. USING THE CURL COMMAND TO VERIFY THAT HTTP/2 IS ENABLED 38
5.5. ADDITIONAL RESOURCES (OR NEXT STEPS) 39

CHAPTER 6. USING A PASSWORD VAULT WITH RED HAT JBOSS WEB SERVER 40
6.1. PASSWORD VAULT INSTALLATION FROM AN ARCHIVE FILE 40
6.2. INSTALLING THE PASSWORD VAULT ON RHEL BY USING THE DNF PACKAGE MANAGER 40
6.3. ENABLING THE PASSWORD VAULT IN JBOSS WEB SERVER 40
6.4. CREATING A JAVA KEYSTORE IN JBOSS WEB SERVER 41
6.5. PASSWORD VAULT INITIALIZATION FOR APACHE TOMCAT 42
6.5.1. Initializing password vault for Apache Tomcat interactively 42
6.5.2. Initializing password vault for Apache Tomcat by using a noninteractive setup 44
6.6. CONFIGURING TOMCAT TO USE THE PASSWORD VAULT 44
6.7. EXTERNAL PASSWORD VAULT CONFIGURATION 45
6.8. STORING A SENSITIVE STRING IN THE PASSWORD VAULT 45
6.9. USING A STORED SENSITIVE STRING IN YOUR TOMCAT CONFIGURATION 46

CHAPTER 7. CONFIGURING THE SSI FILTER 48

CHAPTER 8. CONFIGURING FIPS FOR RED HAT JBOSS WEB SERVER 49
8.1. INTRODUCTION TO FIPS 49
8.2. CONFIGURING FIPS FOR JBOSS WEB SERVER ON RHEL 8 49

APPENDIX A. JAVA IPV4 AND IPV6 PROPERTIES 51
A.1. OVERVIEW OF JAVA IPV4 AND IPV6 PROPERTIES 51
A.2. EXPORTING JAVA IPV4 AND IPV6 PROPERTIES TO TOMCAT 51
A.3. CONFIGURING TOMCAT BINDINGS 51

PROVIDING FEEDBACK ON RED HAT JBOSS WEB SERVER DOCUMENTATION

To report an error or to improve our documentation, log in to your Red Hat Jira account and submit an issue. If you do not have a Red Hat Jira account, then you will be prompted to create an account.

Procedure

1. Click the following link to **create a ticket**.
2. Enter a brief description of the issue in the **Summary**.
3. Provide a detailed description of the issue or enhancement in the **Description**. Include a URL to where the issue occurs in the documentation.
4. Clicking **Submit** creates and routes the issue to the appropriate documentation team.

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

CHAPTER 1. INTRODUCTION TO RED HAT JBOSS WEB SERVER INSTALLATION

Red Hat JBoss Web Server is a fully integrated and certified set of components for hosting Java web applications. Red Hat JBoss Web Server provides a fully supported implementation of the Apache Tomcat Servlet container and the Tomcat native library.

NOTE

If you need clustering or session replication support for Java applications, use Red Hat JBoss Enterprise Application Platform (JBoss EAP).

1.1. JBOSS WEB SERVER COMPONENTS

JBoss Web Server includes components such as the Apache Tomcat Servlet container, Tomcat native library, Tomcat vault, **mod_cluster** library, Apache Portable Runtime (APR), and OpenSSL.

Apache Tomcat

Apache Tomcat is a servlet container in accordance with the Java Servlet Specification. JBoss Web Server 6.x contains Apache Tomcat 10.1.

Tomcat native library

The Tomcat native library improves Tomcat scalability, performance, and integration with native server technologies.

Tomcat vault

Tomcat vault is an extension for JBoss Web Server that is used for securely storing passwords and other sensitive information used by a JBoss Web Server.

Mod_cluster

The **mod_cluster** library enables communication between Apache Tomcat and the **mod_proxy_cluster** module of the Apache HTTP Server. The **mod_cluster** library enables you to use the Apache HTTP Server as a load balancer for JBoss Web Server. For more information about configuring **mod_cluster**, or for information about installing and configuring alternative load balancers such as **mod_jk** and **mod_proxy**, see the HTTP Connectors and Load Balancing Guide .

Apache Portable Runtime

The Apache Portable Runtime (APR) provides an OpenSSL-based TLS implementation for the HTTP connectors.

OpenSSL

OpenSSL is a software library that implements the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols. OpenSSL includes a basic cryptographic library.

For a full list of components that Red Hat JBoss Web Server supports, see the JBoss Web Server Component Details page.

1.2. DIFFERENCES BETWEEN THE APACHE TOMCAT DISTRIBUTIONS THAT RED HAT PROVIDES

Both Red Hat JBoss Web Server and Red Hat Enterprise Linux (RHEL) provide separate distributions of Apache Tomcat. However, JBoss Web Server offers distinct benefits compared to the RHEL distribution of Apache Tomcat by including an integrated and certified set of additional components and features. JBoss Web Server also provides more frequent software and security updates.

NOTE

RHEL provides a distribution of Apache Tomcat on RHEL 7, RHEL 8.8, and RHEL 9.2 or later only.

For RHEL 8.0 through 8.7 and RHEL 9.0 through 9.1, the RHEL platform subscriptions do not provide a distribution of Apache Tomcat. On these operating system versions, JBoss Web Server is the only Apache Tomcat distribution that Red Hat provides, which is available as part of the Middleware Runtimes subscription.

Apache Tomcat versions

Consider the following version information for the Apache Tomcat distributions that are available with JBoss Web Server and RHEL:

- The RHEL 7 **tomcat** package is based on the community version of Apache Tomcat 7.
- The RHEL 8.8 and RHEL 9.x **tomcat** package is based on the community version of Apache Tomcat 9.
- JBoss Web Server 3.1 provides a distribution of Apache Tomcat 7 and Apache Tomcat 8 together with an integrated and certified set of additional components and features. However, Red Hat no longer fully supports or maintains JBoss Web Server 3.1, which is currently in extended life cycle support (ELS) phase 2 with a planned end-of-life date of December 2028.
- JBoss Web Server 5.x provides a distribution of Apache Tomcat 9 that Red Hat fully tests and supports together with an integrated and certified set of additional components and features.
- JBoss Web Server 6.x provides a distribution of Apache Tomcat 10.1 that Red Hat fully tests and supports together with an integrated and certified set of additional components and features.

NOTE

Red Hat does not provide support for community releases of Apache Tomcat.

Differences between JBoss Web Server and RHEL distributions of Apache Tomcat

Consider the following differences between JBoss Web Server and the RHEL distribution of Apache Tomcat:

JBoss Web Server	RHEL distribution of Apache Tomcat
Supports installation from archive files or RPM packages on RHEL versions 8 and 9. NOTE Red Hat does not provide a distribution of JBoss Web Server 6.x on RHEL 7.	Supports installation from RPM packages only on RHEL 7, RHEL 8.8, and RHEL 9.2 or later.
Supports installation from archive files on supported Windows Server platforms.	*Not applicable*

JBoss Web Server	RHEL distribution of Apache Tomcat
Offers developers support for creating and deploying back-end web applications and large-scale websites that can service client requests from Apache HTTP Server proxies in a secure and stable environment.	Offers administrators support for deploying and running Apache Tomcat instances on a RHEL system.

JBoss Web Server	RHEL distribution of Apache Tomcat
Provides a fully tested and supported distribution of Apache Tomcat that includes the following integrated and certified set of additional features and functionality: • Fully tested and certified integration with the Apache HTTP Server for the forwarding and load-balancing of web client requests to back-end web applications by using a **mod_proxy**, **mod_jk**, or **mod_proxy_cluster** connector • Tomcat native library for improving Apache Tomcat scalability, performance, and integration with native server technologies • Tomcat Vault extension for masking passwords and other sensitive strings and securely storing sensitive information in an encrypted Java keystore • **Mod_cluster** library for enabling communication and intelligent load-balancing of web traffic between the **mod_proxy_cluster** module of the Apache HTTP Server and back-end JBoss Web Server worker nodes • Apache Portable Runtime (APR) library for providing an OpenSSL-based TLS implementation for the HTTP connectors • Federal Information Processing Standards (FIPS) compliance • Support for JBoss Web Server in Red Hat OpenShift environments • JBoss Web Server Operator for managing OpenShift container images and for creating, configuring, managing, and seamlessly upgrading instances of web server applications in Red Hat OpenShift environments • Automated installation of JBoss Web Server by using a Red Hat Ansible certified content collection • Offers developers support for creating and deploying back-end web applications and large-scale websites that can service client requests from Apache HTTP Server proxies in a secure and stable environment	Provides only a standard distribution of Apache Tomcat with infrequent software updates that is based on the community version.

JBoss Web Server	RHEL distribution of Apache Tomcat
Provides a set of Maven repository artifacts in a **jws-6.*X.x*-maven-repository.zip** file that you can download from the Red Hat Customer Portal. You can use these artifacts in the web application archive (WAR) files for your application deployment projects.	*Not applicable*
Also includes libraries for embedded Tomcat in the **jws-6.*X.x*-maven-repository.zip** file, which enables you to build web applications by using embedded Tomcat with a fully supported Apache Tomcat version.	*Not applicable*

Differences between the JBoss Web Server and RHEL documentation sets

The JBoss Web Server documentation set is broader and more comprehensive than the RHEL documentation for the **tomcat** package:

- JBoss Web Server includes a **Red Hat JBoss Web Server 6.0.*x* Documentation** archive file that contains API documentation for Apache Tomcat 10.1 and Tomcat Vault. You can download this archive file from the Red Hat Customer Portal .
- The JBoss Web Server product documentation page provides information on all of the following types of use cases:
  - Performing a standard installation of JBoss Web Server from an archive file or RPM package on supported operating systems.
  - Configuring JBoss Web Server for use with Apache HTTP Server connectors and load-balancers such as **mod_jk** and **mod_proxy_cluster**.
  - Enabling automated installations of JBoss Web Server by using a Red Hat Ansible certified content collection.
  - Using JBoss Web Server in a Red Hat OpenShift environment.
  - Installing and using the JBoss Web Server Operator for OpenShift.
  - Configuring JBoss Web Server to support features such as the HTTP/2 protocol, Tomcat Vault, and FIPS compliance.

1.3. JBOSS WEB SERVER OPERATING SYSTEMS AND CONFIGURATIONS

Red Hat JBoss Web Server supports different versions of the Red Hat Enterprise Linux and Microsoft Windows operating systems.

Additional resources

- JBoss Web Server 6 Supported Configurations

1.4. JBOSS WEB SERVER INSTALLATION METHODS

You can install Red Hat JBoss Web Server on supported Red Hat Enterprise Linux and Microsoft Windows systems by using archive installation files that are available for each platform. You can also install JBoss Web Server on supported Red Hat Enterprise Linux systems by using RPM packages.

The following components are included in the archive installation files. These components are the core parts of a JBoss Web Server installation.

- **jws-6.0.0-application-server.zip**
  - Apache Tomcat 10.1
  - **mod_cluster**
  - Tomcat vault
- **jws-6.0.0-optional-native-components-*<platform>*-*<architecture>*.zip**
  - Platform-specific utilities

1.5. JBOSS WEB SERVER COMPONENT DOCUMENTATION BUNDLE

JBoss Web Server includes an additional documentation bundle that includes the original vendor documentation for each component. You can download this documentation bundle, **jws-6.0.0-docs.zip**, from the Red Hat Customer Portal .

The documentation bundle contains additional documentation for the following components:

- Apache Tomcat
- Tomcat native library
- Tomcat vault

CHAPTER 2. INSTALLING JBOSS WEB SERVER ON RED HAT ENTERPRISE LINUX FROM ARCHIVE FILES

You can install JBoss Web Server on Red Hat Enterprise Linux (RHEL) from archive files or RPM packages. If you want to install JBoss Web Server from archive files, you can download and extract the JBoss Web Server archive files from the Red Hat Customer Portal .

When you install JBoss Web Server from an archive file, you can manage the product in different ways. For example, you can use a system daemon at system startup or manage JBoss Web Server from a command line.

NOTE

You can install JBoss Web Server on RHEL versions 8 and 9. Red Hat does not provide a distribution of JBoss Web Server 6.x for RHEL 7 systems.

2.1. PREREQUISITES

- You have installed a supported Java Development Kit (JDK) by using the DNF package manager or from a compressed archive.
- Your system is compliant with Red Hat Enterprise Linux package requirements.

2.1.1. Installing a JDK by using the DNF package manager

You can use the DNF package manager to install a Java Development Kit (JDK). For a full list of supported JDKs, see JBoss Web Server operating systems and configurations .

NOTE

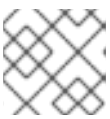
This procedure describes how to install OpenJDK. If you want to install the Oracle JDK, see the Oracle documentation for more information.

Procedure

1. Subscribe your Red Hat Enterprise Linux system to the appropriate channel:
   - rhel-8-server-rpms
   - rhel-9-server-rpms
2. To install a supported JDK version, enter the following command as the root user:

   ```
   # dnf install java-<version>-openjdk-headless
   ```

   In the preceding command, replace **java-*<version>*** with **java-11** or **java-17**.

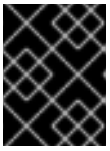
   NOTE

   JBoss Web Server 6.x does not support OpenJDK 8.

3. To ensure the correct JDK is in use, enter the following command as the root user:

```
# alternatives --config java
```

The preceding command returns a list of available JDK versions with the selected version marked with a plus (**+**) sign. If the selected JDK is not the desired one, change to the desired JDK as instructed in the shell prompt.



IMPORTANT

All software that uses the **java** command uses the JDK set by **alternatives**. Changing Java alternatives might impact on the running of other software.

2.1.2. Installing a JDK from a compressed archive

You can install a Java Development Kit (JDK) from a compressed archive such as a **.zip** or **.tar** file. For a full list of supported JDKs, see JBoss Web Server operating systems and configurations .

Procedure

1. If you downloaded the JDK from the vendor's website (Oracle or OpenJDK), use the installation instructions provided by the vendor and set the **JAVA_HOME** environment variable.
2. If you installed the JDK from a compressed archive, set the **JAVA_HOME** environment variable for Tomcat:
   a. In the **bin** directory of Tomcat (***JWS_HOME*/tomcat/bin**), create a file named **setenv.sh**.
   b. In the **setenv.sh** file, enter the **JAVA_HOME** path definition. For example:

```
$ cat JWS_HOME/tomcat/bin/setenv.sh

export JAVA_HOME=/usr/lib/jvm/jre-<version>-openjdk.x86_64
```

   In the preceding example, replace **jre-*<version>*** with **jre-11** or **jre-17**.

2.1.3. Red Hat Enterprise Linux package requirements

Before you install JBoss Web Server on Red Hat Enterprise Linux, you must ensure that your system is compliant with the following package requirements.

- On Red Hat Enterprise Linux version 8 or 9, if you want to use OpenSSL or Apache Portable Runtime (APR), you must install the **openssl** and **apr** packages that Red Hat Enterprise Linux provides.
  - To install the **openssl** package, enter the following command as the root user:

```
# dnf install openssl
```

  - To install the **apr** package, enter the following command as the root user:

```
# dnf install apr
```

- You must remove the **tomcatjss** package before you install the **tomcat-native** package. The **tomcatjss** package uses an underlying Network Security Services (NSS) security model rather than the OpenSSL security model.

To remove the **tomcatjss** package, enter the following command as the root user:

```
# dnf remove tomcatjss
```

2.2. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR A BASE RELEASE OF JBOSS WEB SERVER

A *base release* is the initial release of a specific product version (for example, **6.0.0** is the base release of version 6.0). You can download the JBoss Web Server archive files from the Software Downloads page on the Red Hat Customer Portal.

Prerequisites

- You have installed a supported Java Development Kit (JDK) by using the DNF package manager or from a compressed archive.
- Your system is compliant with Red Hat Enterprise Linux package requirements.

Procedure

1. Open a browser and log in to the Red Hat Customer Portal.
2. Click the **Downloads** tab.
3. From the Product Downloads list, select **Red Hat JBoss Web Server**.
4. On the Software Downloads page, from the **Version** drop-down list, select the appropriate JBoss Web Server version.
5. Click **Download** next to the **Red Hat JBoss Web Server 6.0.0 Application Server** file. The downloaded file is named **jws-6.0.0-application-server.zip** on your local host.
6. If you also want to download the native JBoss Web Server components for your operating system, click **Download** next to the **Red Hat JBoss Web Server 6.0.0 Optional Native Components for *<platform> <architecture>*** file. In this situation, ensure that you select the correct file that matches the platform and architecture for your system. The downloaded file is named **jws-6.0.0-optional-native-components-*<platform>*-*<architecture>*.zip** (for example, **jws-6.0.0-optional-native-components-RHEL8-x86_64.zip**).
7. Extract the downloaded archive files to your installation directory. For example:

```
# unzip jws-6.0.0-application-server.zip -d /opt/
# unzip -o jws-6.0.0-optional-native-compoonents-<platform>-<architecture>.zip -d /opt/
```

The top-level directory for JBoss Web Server is created when you extract the archive. This document refers to the top-level directory for JBoss Web Server as ***JWS_HOME***.

2.3. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR JBOSS WEB SERVER PATCH UPDATES

If product patch updates are available for the appropriate JBoss Web Server version, you can install the archive files for the latest cumulative patches. You can download the JBoss Web Server archive files from the Software Downloads page on the Red Hat Customer Portal.

IMPORTANT

You *cannot* use cumulative patch updates to install the base (*X.X*.0) release of a product version. For example, the installation of a 6.0.2 patch would install the 6.0.1 and 6.0.2 releases but cannot install the base 6.0.0 release.

Service pack releases are cumulative. By downloading the latest service pack release, you also install any previous service pack releases automatically.

Prerequisites

- You have downloaded and extracted the archive files for the base JBoss Web Server release .

Procedure

1. Open a browser and log in to the Red Hat Customer Portal .
2. Click the **Downloads** tab.
3. From the Product Downloads list, select **Red Hat JBoss Web Server**.
4. On the Software Downloads page, from the **Version** drop-down list, select the appropriate JBoss Web Server version.
5. Click the **Patches** tab.
6. Click **Download** next to the latest **Red Hat JBoss Web Server 6.0 Update *XX* Application Server** file.
   The downloaded file is named **jws-6.0.*x*-application-server.zip** on your local host.
7. If you also want to download the native JBoss Web Server components for your operating system, click **Download** next to the latest **Red Hat JBoss Web Server 6.0 Update *XX* Optional Native Components for *<platform> <architecture>*** file. In this situation, ensure that you select the correct file that matches the platform and architecture for your system.
   The downloaded file is named **jws-6.0.*x*-optional-native-components-*<platform>*-*<architecture>*.zip** (for example, **jws-6.0.*x*-optional-native-components-RHEL8-x86_64.zip**).
8. Extract the downloaded archive files to your installation directory.
   For example:

```
# unzip jws-6.0.x-application-server.zip -d /opt/
# unzip -o jws-6.0.x-optional-native-compoonents-<platform>-<architecture>.zip -d /opt/
```

2.4. MANAGING JBOSS WEB SERVER BY USING SYSTEMD WHEN INSTALLED FROM AN ARCHIVE FILE

When you install JBoss Web Server from an archive file on Red Hat Enterprise Linux, you can use a system daemon to perform management tasks. Using the JBoss Web Server with a system daemon provides a method of starting the JBoss Web Server services at system startup. The system daemon

also provides start, stop and status check functions.

On Red Hat Enterprise Linux versions 8 and 9, the default system daemon is **systemd**.

Prerequisites

- You have installed JBoss Web Server from an archive file .

Procedure

1. To determine which system daemon is running, enter the following command:

```
$ ps -p 1 -o comm=
```

   If **systemd** is running, the following output is displayed:

```
systemd
```

2. To set up the JBoss Web Server for **systemd**, run the **.postinstall.systemd** script as the root user:

```
# cd JWS_HOME/tomcat
# sh .postinstall.systemd
```

3. To control the JBoss Web Server with **systemd**, you can perform any of the following steps as the root user:

   - To enable the JBoss Web Server services to start at system startup by using **systemd**:

```
# systemctl enable jws6-tomcat.service
```

   - To start the JBoss Web Server by using **systemd**:

```
# systemctl start jws6-tomcat.service
```

NOTE

The **SECURITY_MANAGER** variable is now deprecated for JBoss Web Server configurations that are based on archive file installations. Consider the following deprecation comment:

```
# SECURITY_MANAGER has been deprecated. To run tomcat under the
Java Security Manager use:
  JAVA_OPTS="-Djava.security.manager -
Djava.security.policy==\"$CATALINA_BASE/conf/"catalina.policy\""""
```

   - To stop the JBoss Web Server by using **systemd**:

```
# systemctl stop jws6-tomcat.service
```

   - To verify the status of the JBoss Web Server by using **systemd**:

```
# systemctl status jws6-tomcat.service
```

NOTE

Any user can run the **status** operation.

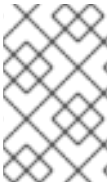
Additional resources

- RHEL 8: Configuring basic system settings: Managing system services with systemctl
- RHEL 9: Configuring basic system settings: Managing system services with systemctl

2.5. JBOSS WEB SERVER CONFIGURATION FOR MANAGING ARCHIVE INSTALLATIONS FROM THE COMMAND LINE

When you install JBoss Web Server from an archive file on Red Hat Enterprise Linux, you can start and stop JBoss Web Server directly from the command line. Before you can run JBoss Web Server from the command line, you must perform the following series of configuration tasks:

- Set the **JAVA_HOME** environment variable for Tomcat.
- Create a **tomcat** user and its parent group.
- Grant the **tomcat** user access to JBoss Web Server.



NOTE

When you manage JBoss Web Server by using a system daemon rather than from the command line, the **.postinstall.systemd** script performs these configuration steps automatically.

2.5.1. Setting the JAVA_HOME environment variable for Apache Tomcat

Before you run JBoss Web Server from the command line for the first time, you must set the **JAVA_HOME** environment variable for Apache Tomcat.

Prerequisites

- You have installed JBoss Web Server from an archive file .

Procedure

1. On a command line, go to the ***JWS_HOME*/tomcat/bin** directory.
2. Create a file named **setenv.sh**.
3. In the **setenv.sh** file, enter the **JAVA_HOME** path definition.
   For example:

```
export JAVA_HOME=/usr/lib/jvm/jre-11-openjdk.x86_64
```

2.5.2. Creating a Tomcat user and group

Before you run JBoss Web Server from the command line for the first time, you must create a **tomcat** user account and user group to enable simple and secure user management. On Red Hat Enterprise Linux, the user identifer (UID) for the **tomcat** user and the group identifier (GID) for the **tomcat** group both have a reserved value of **53**.

NOTE

You must perform all steps in this procedure as the root user.

Prerequisites

- You have set the **JAVA_HOME** environment variable for Tomcat.

Procedure

1. On a command line, go to the ***JWS_HOME*** directory.
2. Create the **tomcat** user group:

```
# groupadd -g 53 -r tomcat
```

3. Create the **tomcat** user in the **tomcat** user group:

```
# useradd -c "tomcat" -u 53 -g tomcat -s /sbin/nologin -r tomcat
```

The preceding commands set both the UID and the GID to **53**. If you subsequently want to change the UID and GID values, see Changing the UID and GID for the **tomcat** user and group.

2.5.3. Granting the Tomcat user access to JBoss Web Server

Before you run JBoss Web Server from the command line for the first time, you must grant the **tomcat** user access to JBoss Web Server by assigning ownership of the Tomcat directories to the **tomcat** user.

NOTE

You must perform all steps in this procedure as the root user.

Prerequisites

- You have created a **tomcat** user and its parent group .

Procedure

1. Go to the ***JWS_HOME*** directory.
2. Assign ownership of the Tomcat directories to the **tomcat** user:

```
# chown -R tomcat:tomcat tomcat/
```

3. Ensure that the **tomcat** user has execute permissions for all parent directories:

```
# chmod -R u+X tomcat/
```

Verification

- Verify that the **tomcat** user is the owner of the directory:

```
# ls -l
```

2.6. STARTING JBOSS WEB SERVER FROM THE COMMAND LINE WHEN INSTALLED FROM AN ARCHIVE FILE

When you install JBoss Web Server from an archive file on Red Hat Enterprise Linux, you can start JBoss Web Server directly from the command line.

Prerequisites

- You have set the **JAVA_HOME** environment variable for Tomcat.
- You have created a **tomcat** user and its parent group.
- You have granted the **tomcat** user access to JBoss Web Server.

Procedure

- Enter the following command as the **tomcat** user:

```
$ sh JWS_HOME/tomcat/bin/startup.sh
```

2.7. STOPPING JBOSS WEB SERVER FROM THE COMMAND LINE WHEN INSTALLED FROM AN ARCHIVE FILE

When you install JBoss Web Server from an archive file on Red Hat Enterprise Linux, you can stop JBoss Web Server directly from the command line.

Prerequisites

- You have started JBoss Web Server from the command line.

Procedure

- Enter the following command as the **tomcat** user:

```
$ sh JWS_HOME/tomcat/bin/shutdown.sh
```

2.8. SELINUX POLICIES FOR JBOSS WEB SERVER

You can use Security-Enhanced Linux (SELinux) policies to define access controls for JBoss Web Server. These policies are a set of rules that determine access rights to the product.

2.8.1. SELinux policy information for jws6-tomcat

The SELinux security model is enforced by the kernel and ensures that applications have limited access to resources such as file system locations and ports. SELinux policies ensure that any errant processes that are compromised or poorly configured are restricted or prevented from running.

The **jws6-tomcat-selinux** packages in your JBoss Web Server installation provide a **jws6_tomcat** policy. The following table contains information about the supplied SELinux policy.

Table 2.1. RPMs and default SELinux policies

Name	Port Information	Policy Information
jws6_tomcat	Four ports in **http_port_t** (TCP ports **8080**, **8005**, **8009**, and **8443**) to allow the tomcat process to use them	The **jws6_tomcat** policy is installed, which sets the appropriate SELinux domain for the process when Tomcat executes. It also sets the appropriate contexts to allow Tomcat to write to the following directories: • **/var/opt/rh/jws6/lib/tomcat** • **/var/opt/rh/jws6/log/tomcat** • **/var/opt/rh/jws6/cache/tomcat** • **/var/opt/rh/jws6/run/tomcat.pid**

Additional resources

- RHEL 8: Using SELinux
- RHEL 9: Using SELinux

2.8.2. Installing SELinux policies for a JBoss Web Server archive installation

In this release, the archive packages provide SELinux policies. The **tomcat** folder of the **jws-6.0.0-application-server-*<platform>*-*<architecture>*.zip** archive includes the **.postinstall.selinux** file. If required, you can run the **.postinstall.selinux** script.

Procedure

1. Install the **selinux-policy-devel** package:

```
dnf install -y selinux-policy-devel
```

2. Run the **.postinstall.selinux** script:

```
cd <JWS_home>/tomcat/
sh .postinstall.selinux
```

3. Add access permissions to the required ports for JBoss Web Server:

```
semanage port -a -t http_port_t -p tcp <port>
```

NOTE

The JBoss Web Server has access to ports **8080**, **8009**, **8443** and **8005** on Red Hat Enterprise Linux systems.

When additional ports are required for JBoss Web Server, use the preceding **semanage** command to provide the necessary permissions, and replace ***<port>*** with the required port.

4. Start Tomcat:

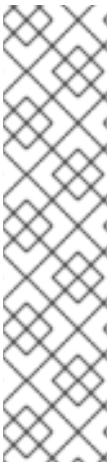
```
<JWS_home>/tomcat/bin/startup.sh
```

5. Check the context of the running process expecting **jws6_tomcat**:

```
ps -eo pid,user,label,args | grep jws6_tomcat | head -n1
```

6. Verify the contexts of the Tomcat directories. For example:

```
ls -lZ <JWS_home>/tomcat/logs/
```

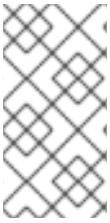


NOTE

By default, the SElinux policy that JBoss Web Server provides is not active and the Tomcat processes run in the **unconfined_java_t** domain. This domain does not confine the processes.

If you choose not to enable the SELinux policy that is provided, you can take the following security measures:

- Restrict file access for the **tomcat** user, so that the **tomcat** user only has access to the files and directories that are necessary for the JBoss Web Server runtime.
- Do not run Tomcat as the root user.



NOTE

When JBoss Web Server is installed from an archive file, Red Hat does not officially support the use of network file sharing (NFS). If you want your JBoss Web Server installation to use an NFS-mounted file system, you are responsible for ensuring that SELinux policies are modified correctly to support this type of deployment.

2.9. CHANGING THE UID AND GID FOR THE TOMCAT USER AND GROUP

On Red Hat Enterprise Linux, the user identifer (UID) for the **tomcat** user and the group identifier (GID) for the **tomcat** group both have a reserved value of **53**. Depending on your setup requirements, you can change the UID and GID for the **tomcat** user and group to some other value.

WARNING

To avoid SELinux conflicts, use UID and GID values that are less than 500. If SELinux is set to **enforcing** mode, UID and GID values greater than 500 might cause unexpected issues.

Prerequisites

- You have created a **tomcat** user account and group .

Procedure

1. If JBoss Web Server is already running, stop JBoss Web Server as the **tomcat** user. For more information, see Stopping JBoss Web Server from the command line when installed from an archive file.
2. To view the current UID and GID for the **tomcat** user and group, enter the following command as the root user:

```
id tomcat
```

   The preceding command displays the user account and group details. For example:

```
uid=53(tomcat) gid=53(tomcat) groups=53(tomcat)
```

3. To assign a new GID to the **tomcat** group, enter the following command as the root user:

```
groupmod -g <new_gid> tomcat
```

   For example:

```
groupmod -g 410 tomcat
```

4. To assign a new UID to the **tomcat** user, enter the following command as the root user:

```
usermod -u <new_uid> -g <new_gid> tomcat
```

   For example:

```
usermod -u 401 -g 410 tomcat
```

5. To reassign file and directory permissions to the new UID, enter the following command as the root user:

```
# find / -not -path '/proc*' -uid <original_uid> | perl -e '$ug = @ARGV[0]; foreach $fn
(<STDIN>) { chomp($fn);$m = (stat($fn))[2];chown($ug,-1,$fn);chmod($m,$fn)}' <new_uid>
```

In the preceding command, replace *<original_uid>* with the old UID and replace *<new_uid>* with the new UID. For example, to reassign file and directory permissions from UID **53** to UID **401**, enter the following command:

```
# find / -not -path '/proc*' -uid 53 | perl -e '$ug = @ARGV[0]; foreach $fn (<STDIN>) {
chomp($fn);$m = (stat($fn))[2];chown($ug,-1,$fn);chmod($m,$fn)}' 401
```

6. To reassign file and directory permissions to the new GID, enter the following command as the root user:

```
# find / -not -path '/proc*' -gid <original_gid> | perl -e '$ug = @ARGV[0]; foreach $fn
(<STDIN>) { chomp($fn);$m = (stat($fn))[2];chown(-1,$ug,$fn);chmod($m,$fn)}' <new_gid>
```

In the preceding command, replace *<original_gid>* with the old GID and replace *<new_gid>* with the new GID. For example, to reassign file and directory permissions from GID **53** to GID **410**, enter the following command:

```
# find / -not -path '/proc*' -gid 53 | perl -e '$ug = @ARGV[0]; foreach $fn (<STDIN>) {
chomp($fn);$m = (stat($fn))[2];chown(-1,$ug,$fn);chmod($m,$fn)}' 410
```

7. To restart JBoss Web Server as the **tomcat** user, see Starting JBoss Web Server from the command line when installed from an archive file.

Additional resources

- What are the reserved UIDs/GIDs in Red Hat Enterprise Linux?

CHAPTER 3. INSTALLING JBOSS WEB SERVER ON RED HAT ENTERPRISE LINUX FROM RPM PACKAGES

You can install JBoss Web Server on Red Hat Enterprise Linux (RHEL) from archive files or RPM packages. If you want to install JBoss Web Server from RPM packages, the installation packages are available from Red Hat Subscription Management.

Installing JBoss Web Server from RPM packages deploys Tomcat as a service and installs Tomcat resources into absolute paths.

NOTE

You can install JBoss Web Server on RHEL versions 8 and 9. Red Hat does not provide a distribution of JBoss Web Server 6.x for RHEL 7 systems.

3.1. PREREQUISITES

- You have installed a supported Java Development Kit (JDK) by using the DNF package or from a compressed archive.
- Your system is compliant with Red Hat Enterprise Linux package requirements.

3.1.1. Installing a JDK by using the DNF package manager

You can use the DNF package manager to install a Java Development Kit (JDK). For a full list of supported JDKs, see JBoss Web Server operating systems and configurations .

NOTE

This procedure describes how to install OpenJDK. If you want to install the Oracle JDK, see the Oracle documentation for more information.

Procedure

1. Subscribe your Red Hat Enterprise Linux system to the appropriate channel:
   - rhel-8-server-rpms
   - rhel-9-server-rpms
2. To install a supported JDK version, enter the following command as the root user:

   ```
   # dnf install java-<version>-openjdk-headless
   ```

   In the preceding command, replace **java-*<version>*** with **java-11** or **java-17**.

   NOTE

   JBoss Web Server 6.x does not support OpenJDK 8.

3. To ensure the correct JDK is in use, enter the following command as the root user:

   ```
   # alternatives --config java
   ```

The preceding command returns a list of available JDK versions with the selected version marked with a plus (**+**) sign. If the selected JDK is not the desired one, change to the desired JDK as instructed in the shell prompt.

IMPORTANT

All software that uses the **java** command uses the JDK set by **alternatives**. Changing Java alternatives might impact on the running of other software.

3.1.2. Installing a JDK from a compressed archive

You can install a Java Development Kit (JDK) from a compressed archive such as a **.zip** or **.tar** file. For a full list of supported JDKs, see JBoss Web Server operating systems and configurations .

Procedure

1. If you downloaded the JDK from the vendor's website (Oracle or OpenJDK), use the installation instructions provided by the vendor and set the **JAVA_HOME** environment variable.
2. If you installed the JDK from a compressed archive, set the **JAVA_HOME** environment variable for Tomcat:
   a. In the **bin** directory of Tomcat (***JWS_HOME*/tomcat/bin**), create a file named **setenv.sh**.
   b. In the **setenv.sh** file, enter the **JAVA_HOME** path definition. For example:

```
$ cat JWS_HOME/tomcat/bin/setenv.sh

export JAVA_HOME=/usr/lib/jvm/jre-<version>-openjdk.x86_64
```

   In the preceding example, replace **jre-*<version>*** with **jre-11** or **jre-17**.

3.1.3. Red Hat Enterprise Linux package requirements

Before you install JBoss Web Server on Red Hat Enterprise Linux, you must ensure that your system is compliant with the following package requirements.

- On Red Hat Enterprise Linux version 8 or 9, if you want to use OpenSSL or Apache Portable Runtime (APR), you must install the **openssl** and **apr** packages that Red Hat Enterprise Linux provides.
  - To install the **openssl** package, enter the following command as the root user:

```
# dnf install openssl
```

  - To install the **apr** package, enter the following command as the root user:

```
# dnf install apr
```

- You must remove the **tomcatjss** package before you install the **tomcat-native** package. The **tomcatjss** package uses an underlying Network Security Services (NSS) security model rather than the OpenSSL security model.
  To remove the **tomcatjss** package, enter the following command as the root user:

```
# dnf remove tomcatjss
```

3.2. ATTACHING SUBSCRIPTIONS TO RED HAT ENTERPRISE LINUX

Before you download and install the RPM packages for JBoss Web Server, you must register your system with Red Hat Subscription Management, and subscribe to the respective Content Delivery Network (CDN) repositories. You can subsequently perform some verification steps to ensure that a subscription provides the required CDN repositories.

Procedure

1. Log in to the Red Hat Subscription Management web page.
2. Click the **Systems** tab.
3. Click the **Name** of the system that you want to add the subscription to.
4. Change from the **Details** tab to the **Subscriptions** tab, and then click **Attach Subscriptions**.
5. Select the check box next to the subscription you want to attach, and then click **Attach Subscriptions**.

Verification

1. Log in to the Red Hat Subscriptions web page.
2. In the **Subscription Name** column, click the subscription that you want to select.
3. Under **Products Provided**, you require both of the following:
   - **JBoss Enterprise Web Server**
   - **Red Hat JBoss Core Services**

Additional resources

- RHEL 8: Performing a Standard RHEL 8 Installation: Registering your system using the Subscription Manager User Interface
- RHEL 9: Performing a Standard RHEL 9 Installation: Registering your system using the Subscription Manager User Interface

3.3. INSTALLING JBOSS WEB SERVER FROM RPM PACKAGES BY USING DNF

You can use the DNF package manager to install JBoss Web Server from RPM packages on Red Hat Enterprise Linux.

Prerequisites

- You have installed a supported Java Development Kit (JDK) .
- Your system is compliant with Red Hat Enterprise Linux package requirements .

- You have attached subscriptions to Red Hat Enterprise Linux .

Procedure

1. To subscribe to the JBoss Web Server CDN repositories for your operating system version, enter the following command:

```
# subscription-manager repos --enable <repository>
```

NOTE

In the preceding command, replace ***<repository>*** with the following values:

- On Red Hat Enterprise Linux 8, replace ***<repository>*** with **jws-6-for-rhel-8-x86_64-rpms**.
- On Red Hat Enterprise Linux 9, replace ***<repository>*** with **jws-6-for-rhel-9-x86_64-rpms**.

2. To install JBoss Web Server, enter the following command as the root user:

```
# dnf groupinstall jws6
```

IMPORTANT

When you install JBoss Web Server from RPM packages, the ***JWS_HOME*** folder is **/opt/rh/jws6/root/usr/share**.

NOTE

- You can install each of the packages and their dependencies individually rather than use the **groupinstall** command. The preferred method is to use **groupinstall**.
- The feature to enable NFS usage by using Software Collection is enabled. For more information about this feature, see the Packaging Guide: Using Software Collections over NFS.

3.4. STARTING JBOSS WEB SERVER WHEN INSTALLED FROM RPMS

When you install JBoss Web Server from RPM packages, you can use the commmand line to start JBoss Web Server. You can subsequently view the output of the service **status** command to verify that Tomcat is running successfully.

Procedure

- Enter the following command as the root user:

```
# systemctl start jws6-tomcat.service
```

NOTE

This is the only supported method of starting JBoss Web Server for an RPM installation.

Verification

- To verify that Tomcat is running, enter the following command as any user:

```
# systemctl status jws6-tomcat.service
```

3.5. STOPPING JBOSS WEB SERVER WHEN INSTALLED FROM RPMS

When you install JBoss Web Server from RPM packages, you can use the command line to stop JBoss Web Server. You can subsequently view the output of the service **status** command to verify that Tomcat is running successfully.

Procedure

- Enter the followng command as the root user:

```
# systemctl stop jws6-tomcat.service
```

Verification

- To verify that Tomcat is no longer running, enter the following command as any user:

```
# systemctl status jws6-tomcat.service
```

3.6. CONFIGURING JBOSS WEB SERVER SERVICES TO START AT SYSTEM STARTUP

When you install JBoss Web Server from RPM packages, you can configure JBoss Web Server services to start at system startup.

Procedure

- Enter the following command:

```
# systemctl enable jws6-tomcat.service
```

3.7. SELINUX POLICIES FOR JBOSS WEB SERVER

You can use Security-Enhanced Linux (SELinux) policies to define access controls for JBoss Web Server. These policies are a set of rules that determine access rights to the product.

3.7.1. SELinux policy information for jws6-tomcat

The SELinux security model is enforced by the kernel and ensures that applications have limited access to resources such as file system locations and ports. SELinux policies ensure that any errant processes that are compromised or poorly configured are restricted or prevented from running.

The **jws6-tomcat-selinux** packages in your JBoss Web Server installation provide a **jws6_tomcat** policy. The following table contains information about the supplied SELinux policy.

Table 3.1. RPMs and default SELinux policies

Name	Port Information	Policy Information
jws6_tomcat	Four ports in **http_port_t** (TCP ports **8080**, **8005**, **8009**, and **8443**) to allow the tomcat process to use them	The **jws6_tomcat** policy is installed, which sets the appropriate SELinux domain for the process when Tomcat executes. It also sets the appropriate contexts to allow Tomcat to write to the following directories: • **/var/opt/rh/jws6/lib/tomcat** • **/var/opt/rh/jws6/log/tomcat** • **/var/opt/rh/jws6/cache/tomcat** • **/var/opt/rh/jws6/run/tomcat.pid**

Additional resources

- RHEL 8: Using SELinux
- RHEL 9: Using SELinux

3.7.2. Enabling SELinux policies for a JBoss Web Server RPM installation

When you install JBoss Web Server from RPM packages, the **jws6-tomcat-selinux** package provides SELinux policies for JBoss Web Server. These packages are available in the JBoss Web Server channel.

Procedure

1. Install the **jws6-tomcat-selinux** package:

```
dnf install -y jws6-tomcat-selinux
```

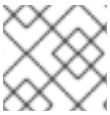
CHAPTER 4. INSTALLING JBOSS WEB SERVER ON MICROSOFT WINDOWS

You can install JBoss Web Server on Microsoft Windows from a set of archive files that you can download from the Red Hat Customer Portal .

4.1. INSTALLING A JDK ON MICROSOFT WINDOWS

Before you install JBoss Web Server on Microsoft Windows, you must first install a Java Development Kit (JDK).

You can download and install the JDK from a supported vendor website, such as Oracle. For a list of supported JDKs, see Supported operating systems and configurations .



NOTE

This procedure describes how to install the Oracle JDK.

Procedure

1. To access the Oracle website, open a browser window and enter the following URL: http://www.oracle.com/technetwork/java/javase/downloads/index.html
2. Download the Oracle JDK for your operating system and architecture.
3. Double-click the downloaded file to start the installation.
4. Proceed as instructed in the installation window.

4.2. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR A BASE RELEASE OF JBOSS WEB SERVER

A *base release* is the initial release of a specific product version (for example, **6.0.0** is the base release of version 6.0). You can download the JBoss Web Server archive files from the Software Downloads page on the Red Hat Customer Portal.

Prerequisites

- You have downloaded and installed a Java Develoment Kit (JDK) from a supported vendor website.

Procedure

1. Open a browser and log in to the Red Hat Customer Portal .
2. Click the **Downloads** tab.
3. From the Product Downloads list, select **Red Hat JBoss Web Server**.
4. On the Software Downloads page, from the **Version** drop-down list, select the appropriate JBoss Web Server version.
5. Click **Download** next to the **Red Hat JBoss Web Server 6.0.0 Application Server** file.

The downloaded file is named **jws-6.0.0-application-server.zip** on your local host.

6. If you also want to download the native JBoss Web Server components for your operating system, click **Download** next to the **Red Hat JBoss Web Server 6.0.0 Optional Native Components for Windows x86_64** file.
   The downloaded file is named **jws-6.0.0-optional-native-components-win6-x86_64.zip**.

7. Extract the downloaded archive files to your installation folder.

The top-level folder for JBoss Web Server is created when you extract the archive. This document refers to the top-level folder for JBoss Web Server as ***JWS_HOME***.

4.3. DOWNLOADING AND EXTRACTING ARCHIVE FILES FOR JBOSS WEB SERVER PATCH UPDATES

If product patch updates are available for the appropriate JBoss Web Server version, you can install the archive files for the latest cumulative patches. You can download the JBoss Web Server archive files from the Software Downloads page on the Red Hat Customer Portal.

IMPORTANT

You *cannot* use cumulative patch updates to install the base (*X.X*.0) release of a product version. For example, the installation of a 6.0.2 patch would install the 6.0.1 and 6.0.2 releases but cannot install the base 6.0.0 release.

Service pack releases are cumulative. By downloading the latest service pack release, you also install any previous service pack releases automatically.

Prerequisites

- You have downloaded and extracted archive files for the appropriate base JBoss Web Server release.

Procedure

1. Open a browser and log in to the Red Hat Customer Portal .
2. Click the **Downloads** tab.
3. From the Product Downloads list, select **Red Hat JBoss Web Server**.
4. On the Software Downloads page, from the **Version** drop-down list, select the appropriate JBoss Web Server version.
5. Click the **Patches** tab.
6. Click **Download** next to the latest **Red Hat JBoss Web Server 6.0 Update *XX* Application Server** file.
   The downloaded file is named **jws-6.0.*x*-application-server.zip** on your local host.
7. If you also want to download the native JBoss Web Server components for your operating system, click **Download** next to the latest **Red Hat JBoss Web Server 6.0 Update *XX* Optional Native Components for Windows x86_64** file.
   The downloaded file is named **jws-6.0.*x*-optional-native-components-win6-x86_64.zip**.

8. Extract the downloaded archive files to your installation folder.

4.4. JBOSS WEB SERVER CONFIGURATION ON MICROSOFT WINDOWS

When you install JBoss Web Server on Microsoft Windows, you can manage JBoss Web Server from a command prompt or by using the Computer Management tool.

Before you can run JBoss Web Server on Microsoft Windows, you must perform the following series of configuration tasks:

- Set environment variables.
- Install the Tomcat service .
- Configure folder permissions.

4.4.1. Setting environment variables for JBoss Web Server on Microsoft Windows

Before you run JBoss Web Server for the first time on Microsoft Windows, you must set the **JAVA_HOME**, **TMP**, and **TEMP** environment variables. You must also update the **PATH** environment variable.

Prerequisites

- You have installed JBoss Web Server.

Procedure

1. Log in to an account with local administrator permissions.
2. Click **Control Panel > System**
3. Click the **Advanced** tab.
4. Click the **Environment Variables** button.
5. Click the **New** button for **System Variables**.
6. For **JAVA_HOME**, **TMP**, and **TEMP**, enter the appropriate name-value pairs for your system.
7. To enable the SSL Connector to work successfully, add ***JWS_HOME*\bin** to the **PATH** environment variable of the user that the services will run under.

NOTE

The services run under the **SYSTEM** user by default.

4.4.2. Installing the Tomcat service on Microsoft Windows

Before you run JBoss Web Server for the first time on Microsoft Windows, you must install the Tomcat service.

Prerequisites

- You have set environment variables for JBoss Web Server.

Procedure

1. Open a command prompt with administrator privileges and go to the **bin** folder for your Tomcat version:

```
cd /D "JWS_HOME\tomcat\bin"
```

2. Install the Tomcat service:

```
call service.bat install
```

4.4.3. Configuring folder permissions for JBoss Web Server services on Microsoft Windows

Before you run JBoss Web Server for the first time on Microsoft Windows, you must configure folder permissions for JBoss Web Server services. Configuring folder permissions ensures that the account that is used to run the JBoss Web Server services has full control over the **JWS_HOME** folder and all of its subfolders.

Prerequisites

- You have installed the Tomcat service on Microsoft Windows .

Procedure

1. Right-click the ***JWS_HOME*** folder and click **Properties**.
2. Select the **Security** tab.
3. Click the **Edit** button.
4. Click the **Add** button.
5. In the text box, enter **LOCAL SERVICE**.
6. Select the **Full Control** check box for the **LOCAL SERVICE** account.
7. Click **OK**.
8. Click the **Advanced** button.
9. Inside the **Advanced Security Settings** dialog, select **LOCAL SERVICE** and click **Change permissions**.
10. Select the check box next to the **Replace all child object permission entries with inheritable permission entries from this object** option.
11. Click **OK** through all the open folder property windows to apply the settings.

4.5. STARTING JBOSS WEB SERVER ON MICROSOFT WINDOWS

When you install JBoss Web Server on Microsoft Windows, you can start the Tomcat service by using the Command Prompt or the Computer Management tool.

Prerequisites

- You have configured JBoss Web Server.

Procedure

- Perform either of the following steps:
  - Open the Command Prompt as an administrator and enter the following command:

```
net start tomcat10
```

  - Click **Start > Administrative Tools > Services**, right-click the **Tomcat10** service, and click **Start**.

NOTE

Some third-party applications add libraries to the system directory in Windows. These third-party libraries take precedence over Tomcat libraries during lookups. If the third-party libraries have the same name as the Tomcat native libraries, the system loads the third-party libraries rather than the libraries that are distributed with JBoss Web Server. In this situation, Tomcat might not start successfully, and Tomcat does not log any error messages in the Windows Event Log or the Tomcat log files.

If this behavior occurs, you can take the following steps:

- To see errors, run the **catalina.bat run** command.
- Inspect the contents of the **C:\windows\System32** directory and the other **PATH** directories.
- Ensure that dynamic link libraries (DLLs) do not conflict with the JBoss Web Server libraries. In particular, look for the **libeay32.dll**, **ssleay32.dll**, and **libssl32.dll** libraries.

4.6. STOPPING JBOSS WEB SERVER ON MICROSOFT WINDOWS

When you install JBoss Web Server on Microsoft Windows, you can stop the Tomcat service by using the Command Prompt or the Computer Management tool.

Prerequisites

- You have started JBoss Web Server on Microsoft Windows.

Procedure

- Perform either of the following steps:
  - Open the Command Prompt as an administrator and enter the following command:

```
net stop tomcat10
```

- Go to **Start > Administrative Tools > Services**, right-click the **Tomcat10** service, and click **Stop**.

CHAPTER 5. ENABLING HTTP/2 FOR THE RED HAT JBOSS WEB SERVER

The Hypertext Transfer Protocols (HTTP) are standard methods of transmitting data between applications, such as servers and browsers, over the internet. JBoss Web Server supports the use of HTTP/2 for encrypted connections that are using Transport Layer Security (TLS), which is indicated by the **h2** keyword when enabled.

HTTP/2 improves on HTTP/1.1 by providing the following enhancements:

- Header compression omits implied information to reduce the size of the header that is transmitted.
- Multiple requests and responses over a single connection use binary framing rather than textual framing to break down response messages.

NOTE

JBoss Web Server does not support the use of HTTP/2 for unencrypted connections that are using the Transmission Control Protocol (TCP), which is indicated by the **h2c** keyword when enabled.

5.1. PREREQUISITES

- You have root user access on Red Hat Enterprise Linux.
- You have installed Red Hat JBoss Web Server 5.0 or later.
- You have installed the **openssl** and **apr** packages that are provided with Red Hat Enterprise Linux. For more information about installing the **openssl** and **apr** packages, see Red Hat Enterprise Linux package requirements.

NOTE

These operating system native libraries are also provided by **jws-6.0.0-application-server-*<platform>*-*<architecture>*.zip** where available.

If you want to run JSSE+OpenSSL or APR on Red Hat Enterprise Linux version 8 or 9, you must use Tomcat-Native to ensure successful operation. Tomcat-Native is located in the native archive directory.

- You have configured a connector that supports the HTTP/2 protocol with SSL enabled. For JBoss Web Server 6.0, the following connectors support the HTTP/2 protocol:
  - The NIO connector with JSSE + OpenSSL (JSSE)
  - The NIO2 connector with JSSE + OpenSSL (JSSE)

5.2. ENABLING HTTP/2 FOR A CONNECTOR

In the **server.xml** file, the upgrade protocol in the connector definition is already set to HTTP/2 by default.

Procedure

1. Open the ***JWS_HOME*/tomcat/conf/server.xml** configuration file.
2. In the connector definition, ensure that the **UpgradeProtocol** class name is set to **org.apache.coyote.http2.Http2Protocol**.
   For example:

```
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
           maxThreads="150" SSLEnabled="true"
           maxParameterCount="1000">
    <UpgradeProtocol className="org.apache.coyote.http2.Http2Protocol" />
    <SSLHostConfig>
        <Certificate certificateKeystoreFile="conf/localhost-rsa.jks"
                     certificateKeystorePassword="changeit"
                     type="RSA" />
    </SSLHostConfig>
</Connector>
```

3. To apply any configuration updates, restart the Red Hat JBoss Web Server as the root user.
   - To restart JBoss Web Server on Red Hat Enterprise Linux by using **systemd**, enter the following command:

```
# systemctl restart jws6-tomcat.service
```

   - To restart JBoss Web Server on Red Hat Enterprise Linux by using **startup.sh**, enter the following commands:

```
# JWS_HOME/sbin/shudown.sh
# JWS_HOME/sbin/startup.sh
```

   - To restart JBoss Web Server on Microsoft Windows, enter the following command:

```
# net restart tomcat10
```

5.3. VIEWING JBOSS WEB SERVER LOGS TO VERIFY THAT HTTP/2 IS ENABLED

You can view the JBoss Web Server console output log to verify that HTTP/2 is enabled.

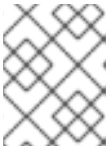
Prerequisites

- You have enabled HTTP/2 for a connector.

Procedure

- To view the console output log, enter the following command:

```
$ cat JWS_HOME/tomcat/logs/catalina.out | grep 'h2'
```



NOTE

In the preceding command, replace ***JWS_HOME*** with the top-level directory for your JBoss Web Server installation.

Verification

- If HTTP/2 is enabled, the command produces the following type of output that indicates the connector has been configured to support negotiation to **[h2]**:

```
06-Apr-2018 04:49:26.201 INFO [main]
org.apache.coyote.http11.AbstractHttp11Protocol.configureUpgradeProtocol The
["connector_name"] connector has been configured to support negotiation to [h2] via ALPN
```

5.4. USING THE CURL COMMAND TO VERIFY THAT HTTP/2 IS ENABLED

You can use the **curl** command-line tool to verify that HTTP/2 is enabled.

Prerequisites

- You have enabled HTTP/2 for a connector.
- You are using a version of **curl** that supports HTTP/2.
  To check that you are using a version of **curl** that supports HTTP/2, enter the following command:

```
$ curl -V
```

  This command produces the following type of output:

```
curl 7.55.1 (x86_64-redhat-linux-gnu) ...
Release-Date: 2017-08-14
Protocols: dict file ftp ftps gopher http https ...
Features: AsynchDNS IDN IPv6 Largefile GSS-API Kerberos SPNEGO NTLM NTLM_WB
SSL libz TLS-SRP HTTP2 UnixSockets HTTPS-proxy Metalink PSL
```

Procedure

1. To check that the HTTP/2 protocol is active, enter the following command:

```
$ curl -I http://<JBoss_Web_Server>:8080/
```

NOTE

In the preceding example, replace ***<JBoss_Web_Server>*** with the URI of the modified connector, such as **example.com**. The port number is dependent on your configuration.

Verification

- If the HTTP/2 protocol is active, the **curl** command produces the following output:

```
HTTP/2 200
```

Otherwise, if the HTTP/2 protocol is inactive, the **curl** command produces the following output:

```
HTTP/1.1 200
```

5.5. ADDITIONAL RESOURCES (OR NEXT STEPS)

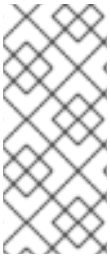
- For more information about using HTTP/2, see Apache Tomcat 10 Configuration Reference: The HTTP Connector - HTTP/2 Support.
- For more information about the HTTP/2 Upgrade Protocol and the supported attributes, see Apache Tomcat 10 Configuration Reference: The HTTP2 Upgrade Protocol .
- For more information about the proposed internet standard for HTTP/2, see IETF: RFC 7540 - Hypertext Transfer Protocol Version 2 (HTTP/2).

CHAPTER 6. USING A PASSWORD VAULT WITH RED HAT JBOSS WEB SERVER

The JBoss Web Server password vault, which is named **tomcat-vault**, is a PicketLink vault extension for Apache Tomcat. You can use the password vault to mask passwords and other sensitive strings, and to store sensitive information in an encrypted Java keystore. When you use the password vault, you can stop storing clear-text passwords in your Tomcat configuration files. Tomcat can use the password vault to search for passwords and other sensitive strings from a keystore.

IMPORTANT

For more information about using the **CRYPT** feature with the password vault, see Using CRYPT.



NOTE

The Federal Information Processing Standard (FIPS) 140-2 does not support the password-based encryption that is provided by **tomcat-vault**. If you want to use password-based encryption on the JBoss Web Server host, you must ensure that FIPS is disabled. If you attempt to use **tomcat-vault** when FIPS mode is enabled, the following error message is displayed: **Security Vault can't be used in FIPS mode**

6.1. PASSWORD VAULT INSTALLATION FROM AN ARCHIVE FILE

When you install JBoss Web Server from an archive file, the password vault is installed automatically when you install the **jws-6.0.0-application-server.zip** file. The password vault is located in the ***JWS_HOME*/tomcat/lib/tomcat-vault.jar** file.

6.2. INSTALLING THE PASSWORD VAULT ON RHEL BY USING THE DNF PACKAGE MANAGER

When you install JBoss Web Server on Red Hat Enterprise Linux from RPM packages, you can use the DNF package manager to install the password vault.

Procedure

- Enter the following command as the root user:

```
dnf install jws6-tomcat-vault
```

6.3. ENABLING THE PASSWORD VAULT IN JBOSS WEB SERVER

You can enable the password vault by adding a configuration property in the **catalina.properties** file.

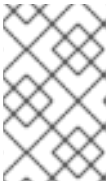
Prequisites

- You have installed the password vault from an archive file or by using the DNF package manager.

Procedure

1. Stop Tomcat if it is already running.
2. Open the ***JWS_HOME*/tomcat/conf/catalina.properties** file.
3. In the **catalina.properties** file, enter the following line:

```
org.apache.tomcat.util.digester.PROPERTY_SOURCE=org.apache.tomcat.vault.util.PropertySourceVault
```



NOTE

In the preceding example, replace ***JWS_HOME*** with the path to your JBoss Web Server installation. The paths shown in this example use a forward slash (/) for directory separators.

6.4. CREATING A JAVA KEYSTORE IN JBOSS WEB SERVER

Before you use the password vault, you must first create a Java keystore by using the **keytool -genseckey** command.

Procedure

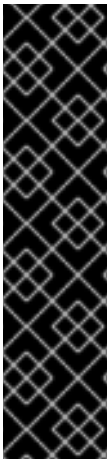
- Enter the following command:

```
$ keytool -genseckey \
 -keystore JWS_HOME/tomcat/vault.keystore \
 -alias my_vault \
 -storetype jceks \
 -keyalg AES \
 -keysize 128 \
 -storepass <vault_password> \
 -keypass <vault_password> \
 -validity 730
```

NOTE

In the preceding example, replace the parameter settings with values that are appropriate for your environment.

For more information about each parameter, use the **keytool -genseckey -help** command.



IMPORTANT

The password vault does not currently support the **PKCS12** keystore type. The password vault supports the **JCEKS** keystore type only.

Depending on the keystore algorithm that you are using, you must specify one of the following **keysize** values:

- If you are using AES, specify **-keysize 128**.
- If you are using DES, specify **-keysize 56**.
- If you are using DESede, specify **-keysize 168**.

6.5. PASSWORD VAULT INITIALIZATION FOR APACHE TOMCAT

You can use the **tomcat-vault.sh** script to initialize the password vault for Apache Tomcat. The **tomcat-vault.sh** script supports either of the following mechanisms to initialize the password vault:

- Interactive setup
- Noninteractive setup

NOTE

Depending on how you installed the password vault, the location of the **tomcat-vault** script varies:

- If you installed the password vault from an archive file, the **tomcat-vault.sh** script is located in the ***JWS_HOME*/tomcat/bin** directory.
- If you installed the password vault by using the DNF package manager, the **tomcat-vault.sh** script is located in the **/opt/rh/jws6/root/usr/bin** directory.

6.5.1. Initializing password vault for Apache Tomcat interactively

You can initialize the password vault for Tomcat interactively. In this situation, the **tomcat-vault.sh** script prompts you to enter values while the script is running.

Procedure

1. Go to the directory that contains the **tomcat-vault.sh** script:
   - If you installed the password vault from an archive file, go to the ***JWS_HOME*/tomcat/bin** directory.
   - If you installed the password vault from an RPM package, go to the **/opt/rh/jws6/root/usr/bin** directory.
2. Run the **tomcat-vault.sh** script:

```
$ ./tomcat-vault.sh
```

3. Follow the on-screen prompts.
   For example:

```
WARNING JBOSS_HOME may be pointing to a different installation - unpredictable results
may occur.

=========================================================================

 JBoss Vault

 JBOSS_HOME: JWS_HOME/tomcat

 JAVA: java

=========================================================================


*********************************
****  JBoss Vault  ***************
*********************************
Please enter a Digit::
0: Start Interactive Session
1: Remove Interactive Session
2: Exit

0

Starting an interactive session
Enter directory to store encrypted files: JWS_HOME/tomcat/
Enter Keystore URL: JWS_HOME/tomcat/vault.keystore
Enter Keystore password: <vault_password>
Enter Keystore password again: <vault_password>
Values match
Enter 8 character salt: 1234abcd
Enter iteration count as a number (Eg: 44): 120
Enter Keystore Alias: my_vault
Initializing Vault
Jun 16, 2018 10:24:27 AM org.apache.tomcat.vault.security.vault.PicketBoxSecurityVault init
INFO: PBOX000361: Default Security Vault Implementation Initialized and Ready
Vault Configuration in tomcat properties file:
********************************************
...
KEYSTORE_URL=JWS_HOME/tomcat/vault.keystore
KEYSTORE_PASSWORD=MASK-3CuP21KMHn7G6iH/A3YpM/
KEYSTORE_ALIAS=my_vault
SALT=1234abcd
ITERATION_COUNT=120
ENC_FILE_DIR=JWS_HOME/tomcat/
...
********************************************
Vault is initialized and ready for use
Handshake with Vault complete
Please enter a Digit::
0: Store a secured attribute
1: Check whether a secured attribute exists
2: Exit

2
```

In the preceding example, replace the specified settings with values that are appropriate for your environment.

4. Note the output for the Tomcat properties file. You need this information when configuring Tomcat to use the password vault.

6.5.2. Initializing password vault for Apache Tomcat by using a noninteractive setup

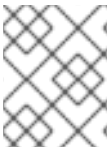
You can initialize the password vault for Tomcat by using a noninteractive setup. In this situation, you must provide the required input as arguments to the **tomcat-vault.sh** script when you run the script.

Procedure

1. Go to the directory that contains the **tomcat-vault.sh** script:
   - If you installed the password vault from an archive file, go to the ***JWS_HOME*/tomcat/bin** directory.
   - If you installed the password vault from an RPM package, go to the **/opt/rh/jws6/root/usr/bin** directory.
2. Run the **tomcat-vault.sh** script and provide the required arguments:
   For example:

```
$ ./tomcat-vault.sh \
 --keystore JWS_HOME/tomcat/vault.keystore \
 --keystore-password <vault_password> \
 --alias my_vault \
 --enc-dir JWS_HOME/tomcat/ \
 --iteration 120 \
 --salt 1234abcd \
 --generate-config JWS_HOME/tomcat/conf/vault.properties
```

   In the preceding example, replace the specified settings with values that are appropriate for your environment.



NOTE

When you specify the **-g, --generate-config** option, the **tomcat-vault.sh** script also creates a **vault.properties** file that contains the specified properties.

6.6. CONFIGURING TOMCAT TO USE THE PASSWORD VAULT

You can configure Apache Tomcat to use the password vault by updating configuration settings in the **vault.properties** file.

Prerequisites

- You have initialized the password vault for Tomcat.

Procedure

1. Go to the ***JWS_HOME*/tomcat/conf/** directory.

2. Create a file named **vault.properties**.

3. In the **vault.properties** file, enter the vault configuration properties that you specified when you initialized the password vault for Tomcat.
   For example:

```
KEYSTORE_URL=JWS_HOME/tomcat/vault.keystore
KEYSTORE_PASSWORD=MASK-3CuP21KMHn7G6iH/A3YpM/
KEYSTORE_ALIAS=my_vault
SALT=1234abcd
ITERATION_COUNT=120
ENC_FILE_DIR=JWS_HOME/tomcat/
```

NOTE

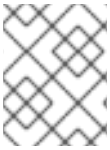
The preceding example is based on the example vault settings in Initializing password vault for Apache Tomcat interactively.

For the **KEYSTORE_PASSWORD** setting, ensure that you use the masked value that was generated when you initialized the password vault.

6.7. EXTERNAL PASSWORD VAULT CONFIGURATION

You can store the **vault.properties** file for the password vault outside of the ***JWS_HOME*/tomcat/conf/** directory. If you have already set a ***CATALINA_BASE*/conf/** directory, you can store the **vault.properties** file in the ***CATALINA_BASE*/conf/** directory.

For more information about setting the ***CATALINA_BASE*** directory, see the "Advanced Configuration - Multiple Tomcat Instances" section in Running The Apache Tomcat 10.1 Servlet/JSP Container on the Apache Tomcat website.



NOTE

The default location for ***CATALINA_BASE*** is ***JWS_HOME*/tomcat/**. This is also known as the ***CATALINA_HOME*** directory.

Additional Resources

- Apache Tomcat 10: Introduction - Directories and Files
- Running The Apache Tomcat 10.1 Servlet/JSP Container : "Advanced Configuration - Multiple Tomcat Instances"

6.8. STORING A SENSITIVE STRING IN THE PASSWORD VAULT

You can use the **tomcat-vault.sh** script to store sensitive strings in the password vault. You can run the **tomcat-vault.sh** script interactively or in a noninteractive mode.

When you add a sensitive string to the password vault, you must specify a name for the string. In this situation, the name of the string is called an **attribute name**, and the string itself is called a **secured attribute**.

Procedure

1. Go to the directory that contains the **tomcat-vault.sh** script:
   - If you installed the password vault from an archive file, go to the ***JWS_HOME*/tomcat/bin** directory.
   - If you installed the password vault from an RPM package, go to the **/opt/rh/jws6/root/usr/bin** directory.
2. To use the **tomcat-vault.sh** script in noninteractive mode, enter the following command:

```
$ ./tomcat-vault.sh \
 --keystore JWS_HOME/tomcat/vault.keystore \
 --keystore-password <vault_password> \
 --alias my_vault \
 --enc-dir JWS_HOME/tomcat \
 --iteration 120 \
 --salt 1234abcd \
 --vault-block my_block \
 --attribute manager_password \
 --sec-attr P@SSW0#D
```

NOTE

The preceding example is based on the example vault settings in Initializing password vault for Apache Tomcat interactively. The preceding example stores the sensitive string, **P@SSW0#D**, with the attribute name, **manager_password**.

When you run the **tomcat-vault.sh** script, you can optionally specify a vault block to store the password in. If you do not specify a block, the **tomcat-vault.sh** script creates a block automatically. The preceding example specifies a vault block named **my_block**.

6.9. USING A STORED SENSITIVE STRING IN YOUR TOMCAT CONFIGURATION

When you store a sensitive string in the password vault, you can refer to the attribute name rather than specify the actual string in your configuration files. By replacing a secured string with the attribute name for the string, you can ensure that the Tomcat configuration file contains only a reference to the password. In this situation, the actual password is stored in the password vault only.

Procedure

1. Open the Tomcat configuration file that contains the sensitive string.
2. Replace the sensitive string with the attribute name for the string, and ensure that you enter the attribute name in the following format: **${VAULT::*block_name*::*attribute_name*::}**
   For example:

   Consider the following example file entry for the secured string, **P@SSW0#D**:

```
<user username="manager" password="P@SSW0#D" roles="manager-gui"/>
```

   If the secured string, **P@SSW0#D**, has the attribute name, **manager_password**, replace the secured string with the following value:

```
<user username="manager" password="${VAULT::my_block::manager_password::}"
roles="manager-gui"/>
```

NOTE

The preceding example is based on the example settings in Storing a sensitive string in the password vault. The preceding example replaces a sensitive string, **P@SSW0#D**, with an attribute name, **manager_password**, that is in a block called, **my_block**.

CHAPTER 7. CONFIGURING THE SSI FILTER

You can configure filter-based Server Side Includes (SSI) support for JBoss Web Server to enable dynamic generation of content in existing HTML pages.

NOTE

SSI directives do not work if you try to configure the SSI filter as in previous versions.

Procedure

1. Open the **conf/web.xml** file.
2. In the **web.xml** file, uncomment the following block:

```
<mime-mapping>
    <extension>shtml</extension>
    <mime-type>text/x-server-parsed-html</mime-type>
</mime-mapping>
```

CHAPTER 8. CONFIGURING FIPS FOR RED HAT JBOSS WEB SERVER

When JBoss Web Server is installed on a Red Hat Enterprise Linux 8 host, you can configure JBoss Web Server to be compliant with Federal Information Processing Standards (FIPS). When you enable FIPS on the Red Hat Enterprise Linux host, this allows JBoss Web Server to operate in FIPS mode automatically.

NOTE

FIPS does not support the password-based encryption functionality that is provided by the **tomcat-vault** component of JBoss Web Server. If you want to use password-based encryption on the JBoss Web Server host, you must ensure that FIPS is disabled. For more information about password-based encryption and **tomcat-vault**, see Vault for Red Hat JBoss Web Server.

8.1. INTRODUCTION TO FIPS

The Federal Information Processing Standards (FIPS) provide guidelines and requirements for improving security and interoperability across computer systems and networks. The FIPS 140-2 and 140-3 series apply to cryptographic modules at both the hardware and software levels. The National Institute of Standards and Technology in the United States implements a cryptographic module validation program with searchable lists of both in-process and approved cryptographic modules.

Red Hat Enterprise Linux provides an integrated framework to enable FIPS 140-2 compliance on a system-wide basis. When operating under FIPS mode, software packages using cryptographic libraries are self-configured according to the global policy.

Additional resources

- Government Standards (Red Hat Customer Portal)
- Security Requirements for Cryptographic Modules (National Institute of Standards and Technology (NIST) website)

8.2. CONFIGURING FIPS FOR JBOSS WEB SERVER ON RHEL 8

You can enable FIPS compliance on the Red Hat Enterprise Linux 8 host during system installation. Alternatively, you can switch your system to FIPS mode after you have completed the system installation.

Procedure

- To enable FIPS mode, complete either of the following steps:
  - If you want to enable FIPS during system installation, follow the instructions in Security Hardening: Installing the system with FIPS mode enabled.
  - If you want to switch to FIPS mode after system installation, follow the instructions in Security Hardening: Switching the system to FIPS mode .

Verification

- Enter the following command:

```
fips-mode-setup --check
```

If FIPS is enabled, this prints the following output:

```
FIPS mode is enabled.
```

APPENDIX A. JAVA IPV4 AND IPV6 PROPERTIES

You can use Java properties to configure IPv4 and IPv6 addresses. You can subsequently export these properties to Tomcat and use address values to specify Tomcat bindings.

A.1. OVERVIEW OF JAVA IPV4 AND IPV6 PROPERTIES

Java provides two properties that you can use to configure IPv4 and IPv6 addresses:

java.net.preferIPv4Stack (default: false)

If IPv6 is available, the underlying native socket is an IPv6 socket by default. This socket enables applications to connect and accept connections from IPv4 and IPv6 hosts. If applications use IPv4 sockets only, set this property to **true**. However, applications that are using IPv4 sockets only cannot communicate with IPv6-only hosts.

java.net.preferIPv6Addresses (default: false)

If a host has both IPv4 and IPv6 addresses, and IPv6 is available, the default behavior is to use IPv4 addresses over IPv6. This allows backward compatibility. If applications depend on an IPv4 address representation, such as 192.168.1.1, set this property to **true** to change the preference, and use IPv6 addresses over IPv4 where possible.

A.2. EXPORTING JAVA IPV4 AND IPV6 PROPERTIES TO TOMCAT

You can export Java IPv4 and IPv6 properties to Tomcat by setting **CATALINA_OPTS** in the ***JWS_HOME*/tomcat/bin/setenv.*** file. On Red Hat Enterprise Linux, the **setenv** file has a **.sh** extension. On Microsoft Windows, the **setenv** file has a **.bat** extension.

Procedure

1. If the ***JWS_HOME*/tomcat/bin/setenv.*** file does not exist, create the file.

NOTE

If you are using Red Hat Enterprise Linux, create a **setenv.sh** file. If you are using Microsoft Windows, create a **setenv.bat** file.

2. To export Java IPv4 and IPv6 properties to Tomcat, perform either of the following steps:

- If you are using Red Hat Enterprise Linux, enter the following command:

```
export "CATALINA_OPTS=-Djava.net.preferIPv4Stack=YOUR_VALUE -
Djava.net.preferIPv6Addresses=YOUR_VALUE"
```

- If you are using Microsoft Windows, enter the following command:

```
set "CATALINA_OPTS=-Djava.net.preferIPv4Stack=YOUR_VALUE -
Djava.net.preferIPv6Addresses=YOUR_VALUE"
```

A.3. CONFIGURING TOMCAT BINDINGS

You can configure Tomcat bindings in the ***JWS_HOME*/tomcat/conf/server.xml** file by specifying the IPv6 address.

Procedure

1. Open the ***JWS_HOME*/tomcat/conf/server.xml** file.
2. To specify the Tomcat binding address, enter the following details:

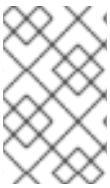
```
<Server ... address="TOMCAT_BINDING_ADDRESS">
```

3. To specify the HTTP connector address, enter the following details:

```
<Connector protocol="HTTP/1.1" ... address="HTTP_CONNECTOR_ADDRESS">
```

4. To specify the AJP connector address, enter the following details:

```
<Connector protocol="AJP/1.3" ... address="AJP_CONNECTOR_ADDRESS">
```



NOTE

Ensure that you replace ***TOMCAT_BINDING_ADDRESS***, ***HTTP_CONNECTOR_ADDRESS***, and ***AJP_CONNECTOR_ADDRESS*** with the correct IPv6 address.