

JBoss Enterprise Web Server 1.0 Installation Guide

for Use with JBoss Enterprise Web Server



Laura Bailey

Eva Kopalová

JBoss Enterprise Web Server 1.0 Installation Guide for Use with JBoss Enterprise Web Server Edition 1.0.2

Author
Author

Laura Bailey
Eva Kopalová

lbailey@redhat.com
ekopalov@redhat.com

Copyright © 2011 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, JBoss, MetaMatrix, 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.

All other trademarks are the property of their respective owners.

This book contains information related to installation and basic configuration of JBoss Enterprise Web Server components. It also includes hardware and platform requirements, and installation prerequisites.

Preface	v
1. Document Conventions	v
1.1. Typographic Conventions	v
1.2. Pull-quote Conventions	vi
1.3. Notes and Warnings	vii
2. We Need Feedback!	vii
1. Introduction	1
1.1. Components	1
1.2. Supported Operating Systems	1
1.3. Supported Connector Configurations	2
1.4. Installation and Upgrade	2
1.5. Source Files	2
I. Installation	3
2. Prerequisites	5
3. Installing Enterprise Web Server on Red Hat Enterprise Linux	7
3.1. Prerequisites	7
3.2. Installing Enterprise Web Server from a ZIP File	9
3.2.1. Downloading and Extracting the ZIP File	9
3.2.2. Configuring the Environment	10
3.2.3. Running Enterprise Web Server	13
3.2.4. Stopping Enterprise Web Server	15
3.2.5. ZIP Installation Structure	15
3.2.6. Additional Resources	15
3.3. Installing Enterprise Web Server from RPM Packages	15
3.3.1. RPM Packages	15
3.3.2. RHN Channels for JBoss Enterprise Web Server Installation Packages	17
3.3.3. Installing Enterprise Web Server from RPM Files	18
3.3.4. Configuring the Environment	19
3.3.5. Setting the Service Boot Behavior	22
3.3.6. Running Enterprise Web Server	22
3.3.7. Stopping Enterprise Web Server	23
3.3.8. Additional Documentation	23
4. Installing Enterprise Web Server on Windows	25
4.1. Prerequisites	25
4.2. Downloading and Extracting Installation File	25
4.3. Configuring the Environment	26
4.4. Running Enterprise Web Server	30
4.5. Stopping Enterprise Web Server	31
4.6. Folder Structure	32
5. Installing Enterprise Web Server on Solaris	33
5.1. Prerequisites	33
5.2. Downloading Installation Files	33
5.3. Extracting and Installing Enterprise Web Server	34
5.4. Configuring the Environment	36
5.5. Running Enterprise Web Server	38
5.5.1. Running Tomcat	39
5.5.2. Running HTTP Server	39
5.6. Stopping Enterprise Web Server	40
5.7. Uninstalling the Package in Solaris	41
5.8. Solaris Directory Structure	41

II. Upgrade	43
6. Upgrading Enterprise Web Server on Red Hat Enterprise Linux	45
6.1. Upgrading ZIP Installation	45
6.2. Upgrading RPM Installation	45
7. Upgrading Enterprise Web Server on Windows	47
8. Upgrading Enterprise Web Server on Solaris	49
9. Hibernate on Enterprise Web Server	51
10. Monitoring Enterprise Web Server with JBoss Operating Network	53
A. Supported HTTP Modules	55
B. Revision History	59

Preface

1. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the [Liberation Fonts](https://fedorahosted.org/liberation-fonts/)¹ set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

1.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

Mono-spaced Bold

Used to highlight system input, including shell commands, file names and paths. Also used to highlight keycaps and key combinations. For example:

To see the contents of the file **my_next_bestselling_novel** in your current working directory, enter the **cat my_next_bestselling_novel** command at the shell prompt and press **Enter** to execute the command.

The above includes a file name, a shell command and a keycap, all presented in mono-spaced bold and all distinguishable thanks to context.

Key combinations can be distinguished from keycaps by the hyphen connecting each part of a key combination. For example:

Press **Enter** to execute the command.

Press **Ctrl+Alt+F2** to switch to the first virtual terminal. Press **Ctrl+Alt+F1** to return to your X-Windows session.

The first paragraph highlights the particular keycap to press. The second highlights two key combinations (each a set of three keycaps with each set pressed simultaneously).

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in **mono-spaced bold**. For example:

File-related classes include **filesystem** for file systems, **file** for files, and **dir** for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialog box text; labeled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

Choose **System** → **Preferences** → **Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, click the **Left-handed mouse** check box and click

¹ <https://fedorahosted.org/liberation-fonts/>

Close to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).

To insert a special character into a **gedit** file, choose **Applications** → **Accessories** → **Character Map** from the main menu bar. Next, choose **Search** → **Find...** from the **Character Map** menu bar, type the name of the character in the **Search** field and click **Next**. The character you sought will be highlighted in the **Character Table**. Double-click this highlighted character to place it in the **Text to copy** field and then click the **Copy** button. Now switch back to your document and choose **Edit** → **Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in proportional bold and all distinguishable by context.

Mono-spaced Bold Italic or ***Proportional Bold Italic***

Whether mono-spaced bold or proportional bold, the addition of italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

To connect to a remote machine using ssh, type **ssh *username@domain.name*** at a shell prompt. If the remote machine is **example.com** and your username on that machine is john, type **ssh *john@example.com***.

The **mount -o remount *file-system*** command remounts the named file system. For example, to remount the **/home** file system, the command is **mount -o remount */home***.

To see the version of a currently installed package, use the **rpm -q *package*** command. It will return a result as follows: ***package-version-release***.

Note the words in bold italics above — *username*, *domain.name*, *file-system*, *package*, *version* and *release*. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

Publican is a *DocBook* publishing system.

1.2. Pull-quote Conventions

Terminal output and source code listings are set off visually from the surrounding text.

Output sent to a terminal is set in **mono-spaced roman** and presented thus:

```
books      Desktop  documentation  drafts  mss    photos  stuff  svn
books_tests Desktop1  downloads      images  notes  scripts  svgs
```

Source-code listings are also set in **mono-spaced roman** but add syntax highlighting as follows:

```
package org.jboss.book.jca.ex1;
import javax.naming.InitialContext;
```

```

public class ExClient
{
    public static void main(String args[])
        throws Exception
    {
        InitialContext iniCtx = new InitialContext();
        Object          ref    = iniCtx.lookup("EchoBean");
        EchoHome        home   = (EchoHome) ref;
        Echo             echo   = home.create();

        System.out.println("Created Echo");

        System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
    }
}

```

1.3. Notes and Warnings

Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.



Note

Notes are tips, shortcuts or alternative approaches to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.



Important

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled 'Important' will not cause data loss but may cause irritation and frustration.



Warning

Warnings should not be ignored. Ignoring warnings will most likely cause data loss.

2. We Need Feedback!

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a report in Bugzilla: <http://bugzilla.redhat.com/bugzilla/> against the product **JBoss Enterprise Web Server**.

When submitting a bug report, be sure to mention the manual's identifier:

JBoss_EWS_Installation_Guide

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

Introduction

JBoss Enterprise Web Server is a fully-integrated and certified set of components for hosting of Java web applications. It is comprised of the web server (Apache HTTP Server), the Apache Tomcat Servlet container, load balancers (mod_jk and mod_cluster), and the Tomcat Native library.

This Installation Guide documents relevant information regarding the installation, minor upgrade, and basic configuration of JBoss Enterprise Web Server on the supported operating systems.

1.1. Components

JBoss Enterprise Web Server consists of the following components:

- **Apache Tomcat** is a servlet container in accordance with Java Servlet Specification. JBoss Enterprise Web Server contains Apache Tomcat 6 and Apache Tomcat 5.
- **Apache Native** is a Tomcat library, which improves Tomcat scalability, performance, and integration with native server technologies. It also contains an implementation of APR (Apache Portable Runtime). APR is a binary protocol for communication of Apache HTTP Server and Apache Tomcat.
- **Apache Tomcat Connectors** (mod_jk, mod_cluster) are connectors between Apache HTTP Server and Apache Tomcat. Note that mod_cluster is a JBoss native load balancer and is more efficient, reliable, and scalable than mod_jk.
- **Apache HTTP Server** is an open-source web server developed by the [Apache Software Foundation](#)¹. The implementation follows the current HTTP standards.



SSL Support

The delivered Apache HTTP Server by default provides the support for the Secure Socket Layer (SSL) protocol. To remove it, refer to the section on removing SSL for your installation.

- **Hibernate** is an object-relational mapping framework. The delivered Hibernate contains Hibernate Core, Hibernate Annotations, Hibernate EntityManager with JPA 1.0 APIs.



Enterprise Web Server is distributed with Red Hat Signed JARs

Red Hat products, including Red Hat Enterprise Linux, are signed with a Red Hat Key to indicate that these packages and JARs are shipped by Red Hat. With Red Hat Signed JARs you can be confident that your product distribution is exactly the distribution that was shipped by Red Hat.



Deprecation of tomcat-jkstatus-ant

Currently, the tomcat-jkstatus-ant library is included in JBoss Enterprise Web Server . However, in the next release, the library will be deprecated and no longer included by default.

1.2. Supported Operating Systems

This release of JBoss Enterprise Web Server is certified for the following operating systems:

- Red Hat Enterprise Linux 6, latest update (x86, x86_64)
- Red Hat Enterprise Linux 5, latest update (x86, x86_64)
- Red Hat Enterprise Linux 4, latest update (x86, x86_64)
- Solaris 10 (x86, x86_64, SPARC64)
- Solaris 9 (x86, SPARC32, SPARC64)
- Windows Server 2008 R2 (x86, x86_64)
- Windows Server 2003 SP2 (x86, x86_64)

1.3. Supported Connector Configurations

This release of JBoss Enterprise Web Server is certified for the following combinations:

- Apache HTTP Server + mod_cluster + JBoss EAP 5.x
- Apache HTTP Server + mod_cluster + Tomcat 6
- Apache HTTP Server + mod_jk + JBoss EAP 5.x
- Apache HTTP Server + mod_jk + Tomcat 5
- Apache HTTP Server + mod_jk + Tomcat 6



The mod_cluster module compatibility

Note that mod_cluster can only be run on Tomcat 6. It is not supported on Tomcat 5.

1.4. Installation and Upgrade

For instructions on how to install JBoss Enterprise Web Server , refer to *Part I, "Installation"* and for instructions on upgrading JBoss Enterprise Web Server , refer to *Part II, "Upgrade"*.

1.5. Source Files

You can obtain the source files at the following locations:

SRPMs

<ftp://ftp.redhat.com/pub/redhat/jbeap/ews/1.0.2/en/source>²

Source ZIP file

<ftp://ftp.redhat.com/pub/redhat/jbews/1.0.2/en/source/jboss-ews-src-1.0.2.zip>³

² <ftp://ftp.redhat.com/pub/redhat/jbeap/ews/1.0.2/en/source>

³ <ftp://ftp.redhat.com/pub/redhat/jbews/1.0.2/en/source/jboss-ews-src-1.0.2.zip>

Part I. Installation

The installation procedure of JBoss Enterprise Web Server differs for various operating systems. Refer to the installation chapter for your operating system:

- for Red Hat Enterprise Linux [Chapter 3, Installing Enterprise Web Server on Red Hat Enterprise Linux](#)
 - for Windows [Chapter 4, Installing Enterprise Web Server on Windows](#)
 - for Solaris [Chapter 5, Installing Enterprise Web Server on Solaris](#)
-

Prerequisites

The following prerequisites must be fulfilled before installing JBoss Enterprise Web Server :

- Enough disk space for the JBoss Enterprise Web Server installation.
- Enough space for your applications.

Installing Enterprise Web Server on Red Hat Enterprise Linux

On Red Hat Enterprise Linux, you can install JBoss Enterprise Web Server in two ways:

- from a ZIP file available on the [Red Hat Customer Portal](#)¹ (refer to [Section 3.2, “Installing Enterprise Web Server from a ZIP File”](#))



ZIP Installation

With a ZIP installation, JBoss Enterprise Web Server can be extracted and installed in any location.

- from RPM packages from Red Hat Network (RHN, refer to [Section 3.2, “Installing Enterprise Web Server from a ZIP File”](#))



RPM Installation

Installation from RPM packages integrates the web server with the **service** command and installs its resources into absolute paths.



RPM Installation and Library Conflicts

RPM installation shares Java library files. Library version conflicts occur when Enterprise Web Server coexists with the Enterprise Application Platform and both were installed from RPM packages. To workaround the issue, install one software with RPM, and the other from a ZIP file.

3.1. Prerequisites

Before installing JBoss Enterprise Web Server on Red Hat Enterprise Linux, ensure the following:

- Supported Java version is installed (refer to [Procedure 3.1, “Installing Java”](#)).
- Required packages are installed (refer to [Procedure 3.2, “Installing Required Packages”](#)).
- The classpathx-jaf package is removed (refer to [Procedure 3.3, “Removing classpathx-jaf”](#))

Procedure 3.1. Installing Java

Follow this procedure to install and set up a supported Java version:

1. Subscribe your system to the appropriate channel:

RHEL Server Supplementary for Red Hat Enterprise Linux 6

RHEL Supplementary for Red Hat Enterprise Linux 5

Extras for Red Hat Enterprise Linux 4

2. On the command line, as the root user run the command to install Java 1.6:

On Red Hat Enterprise Linux 5 and Red Hat Enterprise Linux 6, execute the **yum install** command:

```
# yum install java-1.6.0-<VENDOR>-devel
```

Substitute <VENDOR> with **sun**, **ibm**, or **openjdk**.

On Red Hat Enterprise Linux 4, execute the **up2date** command:

```
# up2date java-1.6.0-<VENDOR>-devel
```

Substitute <VENDOR> with **sun** or **ibm**.

3. Run the following commands as the root user (if switching from another user, make sure to run the command **su -**) to make sure the correct JDK is in use:

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

```
# alternatives --config javac
```

These commands return lists of available JDK versions with the version in use marked with a plus (+) sign. If the set JDK is not the desired option, switch to the right JDK as instructed on the command line.



Important

Any software that makes use of the **java** and **javac** commands is using the JDK set by alternatives. Switching java alternatives may negatively influence running of such software.

4. Install the required packages (refer to [Procedure 3.2, "Installing Required Packages"](#))

Procedure 3.2. Installing Required Packages

If you are using a system registered with Red Hat Network and subscribed to the appropriate channel, you may either mount a repository with the packages or install them from the online repository. Follow this procedure to install the packages from the online repository.

Follow this procedure to install the required packages from the online repository:

1. Depending on your operating system version, run the respective installation command as the root user:

On Red Hat Enterprise Linux 6 and 5:

```
# yum install distcache pcre
```

On Red Hat Enterprise Linux 4:

```
# up2date distcache pcre
```

2. When prompted, press **y** to proceed.

- When the installation is complete, run the following command to check that the packages are installed:

```
# rpm -q distache pcre
```

If the packages are installed, the command output shows the package names with their version numbers.

- Install JBoss Enterprise Web Server either from a ZIP file ([Section 3.2, “Installing Enterprise Web Server from a ZIP File”](#)) or from RPMs ([Section 3.3, “Installing Enterprise Web Server from RPM Packages”](#)).

Procedure 3.3. Removing classpathx-jaf

Follow this procedure to uninstall the *classpathx-jaf* package.

- At the shell prompt, become the root user.
- Depending on your Red Hat Enterprise Linux version, run the respective command to uninstall the package:

On Red Hat Enterprise Linux 5 and 6:

```
# yum remove classpathx-jaf
```

On Red Hat Enterprise Linux 4:

```
# rpm -e classpathx-jaf
```

3.2. Installing Enterprise Web Server from a ZIP File

Make sure that you have installed all the required packages before you install JBoss Enterprise Web Server from a ZIP file (refer to [Procedure 3.2, “Installing Required Packages”](#)).

3.2.1. Downloading and Extracting the ZIP File

To install JBoss Enterprise Web Server, download and extract the installation ZIP file. The ZIP file is available on the [Red Hat Customer Portal](#)².

Procedure 3.4. Downloading the Enterprise Web Server ZIP File

Download the JBoss Enterprise Web Server ZIP file from the [Red Hat Customer Portal](#)³ site:

- Open <http://access.redhat.com> in your web browser.
- In the upper right corner, click the **Log in** button.
- Enter your login information and click **Log in**.
- Hover over the **Downloads** option in the menu bar at the top of the page and under **JBoss Enterprise Middleware** in the displayed menu, click **Downloads**.
- In the **Product** drop-down box column, click **Web Server**.

² <https://access.redhat.com>

³ <https://access.redhat.com>

6. On the Software Downloads page, select the product version in the Version drop-down box and in the **Releases** tab below, click the **JBoss EWS** link with the correct version.
7. On the Software Details page, click the **Download** button next to the File drop-down box.



Note

Make sure the file is intended for your operating system and architecture.

8. Confirm the file download.
9. After the ZIP file has downloaded successfully, follow [Procedure 3.5, “Extracting the ZIP File”](#) to extract the JBoss Enterprise Web Server ZIP file.

Procedure 3.5. Extracting the ZIP File

Unzip the downloaded ZIP file:

1. On the command line, change to the directory with the ZIP file.
2. Run the following **unzip** command:

```
unzip <FILE_NAME> -d <TARGET_DIRECTORY>
```

Substitute *FILE_NAME* with the name of the downloaded file and *TARGET_DIRECTORY* with the path to the target directory.



Note

The target directory and thus the installation directory of the JBoss Enterprise Web Server is referred to as **\$EWS_HOME**. The formatting *<EWS_HOME>* is used to indicate that the entire path to the directory is used; for example *<EWS_HOME>/httpd* could refer to the **/opt/jboss-ews-1.0/httpd** path.

After the ZIP file is extracted, configure the environment properly before running JBoss Enterprise Web Server (refer to [Section 3.2.2, “Configuring the Environment”](#)).

3.2.2. Configuring the Environment

Before running JBoss Enterprise Web Server, the **JAVA_HOME** variable needs to be set (refer to [Procedure 3.6, “Setting JAVA_HOME”](#)).

Optionally, you can also:

- create apache and tomcat users to allow a secure and simple user management (refer to [Procedure 3.8, “Setting the apache User”](#) and [Procedure 3.7, “Creating tomcat User”](#));
- remove the SSL support (refer to [Procedure 3.9, “Removing SSL”](#));
- add log4j logging (refer to [Procedure 3.10, “Configuring log4j”](#));
- enable mod_jk or mod_cluster (refer to [Procedure 3.11, “Configuring mod_jk”](#) and [Procedure 3.12, “Configuring mod_cluster”](#)).

Run the post-installation script once you have set up the environment (refer to [Procedure 3.13](#), “Running the Post-Installation Script”).

Procedure 3.6. Setting JAVA_HOME

Follow this procedure to set the JAVA_HOME variable of your Tomcat to point to a supported Java:

1. In the **bin** directory of your Tomcat (either **\$EWS_HOME/tomcat5/bin** or **\$EWS_HOME/tomcat6/bin**), create the **setenv.sh** file (for example, issue the command **vim tomcat<VERSION>/bin/setenv.sh**).
2. Add the JAVA_HOME path definition into the file; for example **export JAVA_HOME=/usr/lib/jvm/jre-1.6.0-openjdk.x86_64**

Procedure 3.7. Creating tomcat User

Follow this procedure to create the tomcat user and its parent group:

1. At the shell prompt, become the root user.
2. Run the following command to create the tomcat user group:

```
# groupadd -g 91 -r tomcat
```

3. Go to **\$EWS_HOME**.
4. From **\$EWS_HOME**, run the following command to create the tomcat user in the tomcat user group:

```
# useradd -c "Tomcat" -u 91 -g tomcat -s /bin/sh -r -d /home/tomcat tomcat
```

5. From **\$EWS_HOME**, run the following command to assign the ownership of the tomcat directories to the tomcat user and allow the user to run the tomcat service:

```
# chown -R tomcat:tomcat tomcat<VERSION>
```

Substitute **<VERSION>** with the respective tomcat version number (**5** or **6**).

6. From **\$EWS_HOME**, issue the command **ls -l** and check if the tomcat user is the owner of the Tomcat directory.
7. Make sure that the tomcat user has the execution permission to all parent directories.

Procedure 3.8. Setting the apache User

Follow this procedure to set up the apache user:

1. Run the following commands as the root user to create the apache user group:

```
# groupadd -g 48 -r apache
```

2. Run the following command to create the apache user in the apache user group:

```
# useradd -c "Apache" -u 48 -g apache -s /bin/sh -r -d /home/apache apache
```

3. Change to **\$EWS_HOME** and run the following command to assign the ownership of the apache directories to the apache user and allow the user to run **httpd**:

```
# chown -R apache:apache httpd
```

4. From `$EWS_HOME`, issue the command `ls -l` and check that the apache user has execution permission to the respective Apache install path. Output is similar to the following:

```
drwxrwxr-- 11 apache apache 4096 Feb 14 06:52 httpd
```

Procedure 3.9. Removing SSL

JBoss Enterprise Web Server supports SSL by default, but it can be removed.

Follow this procedure to remove or re-add SSL:

1. Go to the `$EWS_HOME/httpd/conf.d/` directory.
2. Rename the SSL configuration file:

To remove SSL, rename `ssl.conf` to `ssl.conf.disabled`.

To re-add SSL, rename `ssl.conf.disabled` to `ssl.conf`.

Procedure 3.10. Configuring log4j

Follow this procedure to add `log4j` logging to Tomcat:

1. Go to the `$EWS_HOME/extras/` directory.
2. Copy the `log4j.jar` and `log4j.properties` files to the `lib` directory of the Tomcat directory:

If using Tomcat 6, run the following commands:

```
extras]# cp log4j.jar log4j.properties ../tomcat6/lib
```

```
extras]# cp tomcat-juli-adapters.jar $EWS_HOME/tomcat6/lib
```

If using Tomcat 5, run the following commands:

```
extras]# cp log4j.properties ../tomcat5/common/classes
```

```
extras]# cp log4j.jar ../tomcat5/common/lib
```

Procedure 3.11. Configuring mod_jk

To configure the HTTP Server to use `mod_jk`, define the following:

- available workers (JBoss instances) in the `workers.properties` file
- the `mod_jk` configuration file

Follow this procedure to configure the HTTP Server to use `mod_jk` as its load balancer:

1. In the `$EWS_HOME/httpd/conf/` directory, create `workers.properties`
2. In the `$EWS_HOME/httpd/conf.d/` directory, create `mod_jk.conf`.



Configuration Samples

You can also use the template files from the `jboss-ews-docs-1.0.2.zip` file (the file is available at the [Red Hat Customer Portal](#)⁴): after you have downloaded and unzipped the file, copy the sample files `mod_jk.conf.sample` and `workers.properties.sample` from the `$EWS_HOME/doc/mod_jk/` to the locations defined in [Step 1](#) and [Step 2](#). Rename them (drop the `sample` extension) and modify their content as needed (to refer to [Section 3.2.6](#), “[Additional Resources](#)”).

Procedure 3.12. Configuring `mod_cluster`

Follow this procedure to configure the HTTP Server to use `mod_cluster` as its load balancer:

1. In the `<EWS_HOME>/httpd/conf/httpd.conf` file, add the hash (#) sign at the beginning of the following line to disable `mod_proxy_balancer.so`:

```
LoadModule proxy_balancer_module /modules/mod_proxy_balancer.so
```

This module is incompatible with the JBoss HTTP Connector.

2. Configure the server to load the JBoss HTTP Connector modules:
 - a. In the `$EWS_HOME/httpd/conf.d/` directory, create the `JBoss_HTTP.conf` file.
 - b. Add the following lines to the `JBoss_HTTP.conf` file:

```
LoadModule slotmem_module <EWS_HOME>/httpd/modules/mod_slotmem.so
LoadModule manager_module <EWS_HOME>/httpd/modules/mod_manager.so
LoadModule proxy_cluster_module <EWS_HOME>/httpd/modules/mod_proxy_cluster.so
LoadModule advertise_module <EWS_HOME>/httpd/modules/mod_advertise.so
```

Procedure 3.13. Running the Post-Installation Script

Follow this procedure to run the post-installation script:

1. At the shell prompt, become the root user.
2. Change to the `$EWS_HOME/httpd` directory.
3. Run the following command:

```
# ./postinstall
```

3.2.3. Running Enterprise Web Server

To run JBoss Enterprise Web Server, run the following:

- Tomcat (5 or 6, refer to [Section 3.2.3.1](#), “[Running Tomcat](#)”)
- HTTP server ([Section 3.2.3.2](#), “[Running HTTP](#)”)

3.2.3.1. Running Tomcat

You can run Tomcat either in the background as a service (refer to [Section 3.2.3.1.2, “Running Tomcat as a Service”](#)) or in the foreground as a program (refer to [Section 3.2.3.1.1, “Running Tomcat as a Program”](#)).



Supported Method of Running Tomcat

It is strongly recommended to run Tomcat with the startup.sh script (refer to the startup.sh entry in [Section 3.2.3.1.1, “Running Tomcat as a Program”](#)). Note that this is the only supported running method.

Before starting Tomcat, make sure the following prerequisites are met:

- The tomcat user is created (refer to [Procedure 3.7, “Creating tomcat User”](#)).
- JAVA_HOME is set correctly (refer to [Procedure 3.6, “Setting JAVA_HOME”](#)).

3.2.3.1.1. Running Tomcat as a Program

To run Tomcat as a program, run one of the following scripts:

- **startup.sh** script

Run the following command as the root user with the respective Tomcat version (**5** or **6**):

```
# sh <EWS_HOME>/tomcat<VERSION>/bin/startup.sh
```

The system reads the configuration file and changes to the user and group to the tomcat user account (that is, Tomcat is always run under the tomcat user). By default, **tomcat-native** is automatically added to the LD_LIBRARY_PATH and -Djava.library.path during Tomcat startup.

- **catalina.sh** script

Run the following command with the respective Tomcat version (**5** or **6**) :

```
# sh <EWS_HOME>/tomcat<VERSION>/bin/catalina.sh
```

- custom script, which calls **catalina.sh**

3.2.3.1.2. Running Tomcat as a Service

To run Tomcat as a service, create and use your own system daemon script with jsvc or tanukiwrapper.

3.2.3.2. Running HTTP

To start httpd, change to **<EWS_HOME>/httpd/sbin/** and run the following command as the root user:

```
# ./apachectl start
```

3.2.4. Stopping Enterprise Web Server

Stop the tomcat and httpd service to stop JBoss Enterprise Web Server (refer to [Section 3.2.4.1, “Stopping Tomcat”](#) and [Section 3.2.4.2, “Stopping HTTP”](#)).

3.2.4.1. Stopping Tomcat

Run the following command as the root user to stop tomcat:

```
# sh <EWS_HOME>/tomcat<VERSION>/bin/shutdown.sh
```

3.2.4.2. Stopping HTTP

To stop httpd, change to `<EWS_HOME>/httpd/sbin/` and run the following command as the root user:

```
# apachectl stop
```

3.2.5. ZIP Installation Structure

The following is the default directory layout after a successful ZIP installation:

```
`-- jboss-ews-1.0
   |-- extras (includes log4j)
   |-- httpd (includes modules, configuration files, executable for the HTTP Server)
   |-- tomcat5
   `-- tomcat6
```

3.2.6. Additional Resources

To acquire the product documentation and source packages, download the **jboss-ews-docs-1.0.2.zip** and **jboss-ews-src-1.0.2.zip** files available at the [Red Hat Customer Portal](#)⁵ and extract them to `$EWS_HOME`. On extraction, **doc** and **src** directories are created, which contain the documentation and examples, and source code.

3.3. Installing Enterprise Web Server from RPM Packages

RPM installation integrates the JBoss Enterprise Web Server with the system and Tomcat and HTTP Server are installed as system services (you can run them as services from the command line).

Before installing JBoss Enterprise Web Server from RPMs, make sure the required packages and a supported Java is installed (refer to [Procedure 3.2, “Installing Required Packages”](#) and [Procedure 3.1, “Installing Java”](#)). Also ensure the system is registered with Red Hat Network and subscribed to the appropriate JBoss Enterprise Web Server channels (refer to [Procedure 3.14, “Subscribing to Red Hat Network Enterprise Channels”](#)).

3.3.1. RPM Packages

JBoss Enterprise Web Server components are distributed in RPM packages for individual Red Hat Enterprise Linux versions.

⁵ <https://access.redhat.com>

RPM Packages for Red Hat Enterprise Linux 5 and 6

Tomcat 6:

tomcat6

tomcat6-webapps

tomcat6-admin-webapps

Tomcat 5:

tomcat5

tomcat5-webapps

tomcat5-admin-webapps

Tomcat Native:

tomcat-native

HTTP Server:

httpd

mod_jk:

mod_jk-ap20

mod_cluster:

mod_cluster-tomcat6

mod_cluster-native

log4j:

tomcat6-log4j

mod_ssl:

mod_ssl

Hibernate:

hibernate3

JK Status Ant tasks:

tomcat-jkstatus-ant



Deprecation of tomcat-jkstatus-ant

In the next release, the tomcat-jkstatus-ant library will be deprecated.

RPM Packages for Red Hat Enterprise Linux 4

Tomcat 6:

tomcat6

tomcat6-webapps

tomcat6-admin-webapps

Tomcat 5:

tomcat5

tomcat5-webapps

tomcat5-admin-webapps

Tomcat Native:

tomcat-native

HTTP Server:

httpd22

mod_jk:

mod_jk-ap20

mod_cluster:

mod_cluster-tomcat6

mod_cluster-native

log4j:

tomcat6-log4j

mod_ssl:

mod_ssl22

Hibernate:

hibernate3

JK Status Ant tasks:

tomcat-jkstatus-ant



Deprecation of tomcat-jkstatus-ant

In the next release, the tomcat-jkstatus-ant library will be deprecated.

Xerces2 Java Parser:

xerces-j2



Xerces2 Java Parser

Xerces2 Java Parser is a Tomcat 5 dependency. If you are using Tomcat 6, the package is installed automatically as a dependency. However, if you are installing only the tomcat5 package, make sure to install xerces-j2 before.

3.3.2. RHN Channels for JBoss Enterprise Web Server Installation Packages

To be able to download and install JBoss Enterprise Web Server RPM packages, you need to subscribe to the channels with the resources intended for your operating system version and architecture (refer to [Section 3.3.3, "Installing Enterprise Web Server from RPM Files"](#)):

- on Red Hat Enterprise Linux 6:
 - jboss-ews-1-i386-server-6-rpm

- jboss-ews-1-x86_64-server-6-rpm
- on Red Hat Enterprise Linux 5:
 - jboss-ews-1-i386-server-5-rpm
 - jboss-ews-1-x86_64-server-5-rpm
- on Red Hat Enterprise Linux 4:
 - jboss-ews-1-i386-as-4-rpm
 - jboss-ews-1-x86_64-as-4-rpm
 - jboss-ews-1-i386-es-4-rpm
 - jboss-ews-1-x86_64-es-4-rpm

3.3.3. Installing Enterprise Web Server from RPM Files

Before downloading and installing the RPM packages, register your system with Red Hat Network and subscribe to the respective channels.

Procedure 3.14. Subscribing to Red Hat Network Enterprise Channels

1. Go to [Red Hat Network Enterprise](#)⁶.
2. Unsubscribe your system from the Red Hat Application Stack channel.
3. Subscribe to the JBoss Enterprise Web Server channels (refer to [Section 3.3.2, “RHN Channels for JBoss Enterprise Web Server Installation Packages”](#)).
4. Ensure that `exactarch` is set to the correct value:

On Red Hat Enterprise Linux 4, skip this step and proceed to [Procedure 3.15, “Installing Enterprise Web Server from RPM Packages”](#).

On Red Hat Enterprise Linux 6 and Red Hat Enterprise Linux 5, open `/etc/yum.conf` for editing and set the `exactarch` value to 0:

```
exactarch=0
```

If the directive is not present, add it to the file.

Procedure 3.15. Installing Enterprise Web Server from RPM Packages

Install the respective packages (refer to [Section 1.1, “Components”](#) and to [Section 3.3.1, “RPM Packages”](#) to identify the packages relevant for your installation):

1. At the shell prompt, become the root user.
2. Depending on your operating system, run the installation command:

On Red Hat Enterprise Linux 6 and Red Hat Enterprise Linux 5:

```
# yum install <LIST_OF_PACKAGES>
```

Substitute the `<LIST_OF_PACKAGES>` with names of the packages separated with spaces.

To install all JBoss Enterprise Web Server packages, run this command:

```
# yum install tomcat6 tomcat6-webapps tomcat6-admin-webapps
tomcat5.noarch tomcat5-webapps.noarch tomcat5-admin-webapps.noarch tomcat-native
httpd mod_jk-ap20 mod_cluster-tomcat6 mod_cluster-native tomcat6-log4j mod_ssl
hibernate3 tomcat-jkstatus-ant
```



Fetching Tomcat 5 Packages

Unless *package.noarch* is specified for Tomcat 5 packages, yum fetches all Tomcat 5 packages from the base channel instead of the JBoss Enterprise Web Server channel.

On Red Hat Enterprise Linux 4:

```
# up2date <LIST_OF_PACKAGES>
```

Substitute the *<LIST_OF_PACKAGES>* with names of the packages divided by spaces.

To install all JBoss Enterprise Web Server packages, run this command:

```
# up2date tomcat6 tomcat6-webapps tomcat6-admin-webapps xerces-j2
tomcat5 tomcat5-webapps tomcat5-admin-webapps tomcat-native httpd22 mod_jk-ap20
mod_cluster-tomcat6 mod_cluster-native tomcat6-log4j mod_ssl22 hibernate3 tomcat-
jkstatus-ant
```

3. Ensure that *exactarch* is set to the correct value:

On Red Hat Enterprise Linux 4, skip this step.

On Red Hat Enterprise Linux 6 and Red Hat Enterprise Linux 5, open `/etc/yum.conf` for editing and set the **exactarch** value to 1:

```
exactarch=1
```

4. Run the following command and verify the system is using the correct javac and java (for details refer to [Procedure 3.1, "Installing Java"](#)).

3.3.4. Configuring the Environment

Before running JBoss Enterprise Web Server, you can optionally configure environment:

- remove SSL (refer to [Procedure 3.16, "Removing SSL"](#));
- add log4j logging (refer to [Procedure 3.18, "Removing log4j From Tomcat 6"](#));
- enable mod_jk or mod_cluster (refer to [Procedure 3.20, "Configuring mod_jk"](#) and [Procedure 3.21, "Configuring mod_cluster"](#))

Procedure 3.16. Removing SSL

The SSL support is optional. It is provided by the mod_ssl package and is enabled on package installation. To remove SSL, do the following:

1. At the shell prompt, become the root user.
2. Run the respective uninstall command:

on Red Hat Enterprise Linux 6 and Red Hat Enterprise Linux 5:

```
# yum remove mod_ssl
```

on Red Hat Enterprise Linux 4:

```
# rpm -e mod_ssl22
```

Configuring log4j

For tomcat6, the log4j support is provided by the *tomcat6-log4j* package and is enabled on package installation (refer to [Procedure 3.15, “Installing Enterprise Web Server from RPM Packages”](#)). To enable log4j on tomcat5, copy the respective log4j resources from the **tomcat6** to the **tomcat5** directory (refer to [Procedure 3.17, “Adding log4j on tomcat5”](#))

Procedure 3.17. Adding log4j on tomcat5

1. Install the **tomcat6-log4j** RPM package.
2. From `/usr/share/java/tomcat6/`, copy:
 - **log4j.jar** to `/usr/share/tomcat5/common/lib` and
 - **log4j.properties** to `/usr/share/tomcat5/common/classes`.

Procedure 3.18. Removing log4j From Tomcat 6

Follow this procedure to remove log4j from Tomcat 6:

1. At the shell prompt, become the root user.
2. Run the respective uninstall command:

on Red Hat Enterprise Linux 6 and Red Hat Enterprise Linux 5:

```
# yum remove tomcat6-log4j
```

on Red Hat Enterprise Linux 4:

```
# rpm -e tomcat6-log4j
```

Procedure 3.19. Removing log4j From Tomcat 5

Follow this procedure to remove log4j from Tomcat 5

1. Remove the **log4j.jar** file from the `/usr/share/tomcat5/common/lib` directory.
2. Remove the **log4j.properties** file from the `/usr/share/tomcat5/common/classes` directory.

Procedure 3.20. Configuring mod_jk

Follow this procedure to configure and enable *mod_jk*:

1. Install the template **mod_jk** configuration file:

```
# cp /usr/share/doc/mod_jk-ap20-*/mod_jk.conf.sample /etc/httpd/conf.d/mod_jk.conf
```

2. Install the template **workers.properties** file:

```
# cp /usr/share/doc/mod_jk-ap20-*/workers.properties.sample /etc/httpd/conf/workers.properties
```

3. Add worker mappings to **workers.properties**. For example:

```
# Mount the Servlet context to the ajp20 worker
/jmx-console=loadbalancer
/jmx-console/*=loadbalancer
/web-console=loadbalancer
/web-console/*=loadbalancer
```



Note

The default configuration is adequate for low-traffic web sites. For optimization information for moderate and high-load web sites, refer to <http://kbase.redhat.com/faq/docs/DOC-15836>.

Procedure 3.21. Configuring mod_cluster

The `mod_cluster` load balancer is optional. It is provided by the `mod_cluster` package and is enabled on package installation.

Follow this procedure to configure the HTTP Server to use `mod_cluster` as its load balancer:

1. In the `/etc/httpd/conf/httpd.conf` file, add the hash (#) sign at the beginning of the following line to disable `mod_proxy_balancer.so`:

```
LoadModule proxy_balancer_module /modules/mod_proxy_balancer.so
```

This module is incompatible with the JBoss HTTP Connector.

2. Configure the server to load the JBoss HTTP Connector modules:

- a. Create the file `/etc/httpd/conf.d/JBoss_HTTP.conf`.

- b. Add the following lines to the **JBoss_HTTP.conf** file:

- for 32-bit architecture:

```
LoadModule slotmem_module /usr/lib/httpd/modules/httpd/modules/mod_slotmem.so
LoadModule manager_module /usr/lib/httpd/modules/mod_manager.so
LoadModule proxy_cluster_module /usr/lib/httpd/modules/mod_proxy_cluster.so
LoadModule advertise_module /usr/lib/httpd/modules/mod_advertise.so
```

- for 64-bit architecture:

```
LoadModule slotmem_module /usr/lib64/httpd/modules/httpd/modules/mod_slotmem.so
LoadModule manager_module /usr/lib64/httpd/modules/mod_manager.so
LoadModule proxy_cluster_module /usr/lib64/httpd/modules/mod_proxy_cluster.so
LoadModule advertise_module /usr/lib64/httpd/modules/mod_advertise.so
```

3.3.5. Setting the Service Boot Behavior

You can define the service behavior (for the tomcat and httpd service) at boot from the command line or with the Service Configuration graphical tool:

- From the command line, issue the following command with the `<SERVICE_NAME>` replaced with the respective service name (**tomcat6** or **tomcat5**, and **httpd** or **httpd22**)

```
chkconfig <SERVICE_NAME> on
```

For details refer to the `chkconfig` man page.

- To use the Service Configuration tool, issue the command **system-config-services** and change the service setting in the displayed window.

3.3.6. Running Enterprise Web Server

To run JBoss Enterprise Web Server , run the following:

- Tomcat (5 or 6, refer to [Procedure 3.22, “Running Tomcat”](#))
- HTTP server ([Procedure 3.23, “Running HTTP”](#))

Procedure 3.22. Running Tomcat



Important

Both Tomcat 5 and Tomcat 6 can be run in parallel on your system. However, to avoid conflicts each version should use its own port address.

Follow this procedure to start **tomcat**:

1. At the shell prompt, become the root user.
2. Run the following **start** command with the `<VERSION>` substituted with the correct Tomcat version (**5** or **6**):

```
# service tomcat<VERSION> start
```

3. Go to <http://localhost:8080> in your web browser to verify that Tomcat is running. The Apache Software Foundation page appears.



Running Tomcat

In JBoss Enterprise Web Server installations from the RPM packages, the only supported way of running Tomcat is the running described in this section.

Procedure 3.23. Running HTTP

Follow this procedure to start **httpd**:

1. At the shell prompt, become the root user.

2. Run the following **start** command:

On Red Hat Enterprise Linux 6 or Red Hat Enterprise Linux 5:

```
# service httpd start
```

On Red Hat Enterprise Linux 4:

```
# service httpd22 start
```

3.3.7. Stopping Enterprise Web Server

To stop JBoss Enterprise Web Server, stop the HTTP Server and Tomcat (refer to [Procedure 3.24, “Stopping Tomcat”](#) and [Procedure 3.25, “Stopping HTTP”](#)).

Procedure 3.24. Stopping Tomcat

Follow this procedure to stop **tomcat**:

1. At the shell prompt, become the root user.
2. Run the following **start** command with the `<VERSION>` substituted with the correct Tomcat version (**5** or **6**):

```
# service tomcat<VERSION> stop
```

3. Go to <http://localhost:8080> in your web browser to verify that Tomcat is no longer running.

Procedure 3.25. Stopping HTTP

Follow this procedure to stop **httpd**:

1. At the shell prompt, become the root user.
2. Run the following **stop** command:

On Red Hat Enterprise Linux 6 or Red Hat Enterprise Linux 5:

```
# service httpd stop
```

On Red Hat Enterprise Linux 4:

```
# service httpd22 stop
```

3.3.8. Additional Documentation

To install additional documentation, issue the command **up2date** in Red Hat Enterprise Linux 4 and **yum install** in Red Hat Enterprise Linux 5 and Red Hat Enterprise Linux 6 followed by the list of required documentation packages.

- The following documentation packages are available for Red Hat Enterprise Linux 6 and Red Hat Enterprise Linux 5:

for Tomcat 5:

tomcat5-webapps

for Tomcat 6:

mod_jk-manual

httpd-manual

tomcat6-docs-webapp

- The following documentation packages are available for Red Hat Enterprise Linux 4:

for Tomcat 5:

tomcat5-webapps

for Tomcat 6:

mod_jk-manual

httpd22-manual

tomcat6-docs-webapp

Installing Enterprise Web Server on Windows

Follow the procedures in this chapter to install, configure, and run JBoss Enterprise Web Server on Windows.

4.1. Prerequisites

Before installing JBoss Enterprise Web Server on Windows, confirm that **Java SDK 1.6** is installed (refer to [Procedure 4.1, “Installing Java”](#)).

Procedure 4.1. Installing Java

1. Go to the [Oracle homepage](#)¹.
2. Download JDK 1.6 for your operating system from the web site.
3. Go to the download location.
4. Double-click the downloaded file to start the Java installation.
5. Proceed as instructed in the displayed installation window.

4.2. Downloading and Extracting Installation File

Download the JBoss Enterprise Web Server ZIP file from the [Red Hat Customer Portal](#)² site that is correct for your architecture.

Procedure 4.2. Downloading the Enterprise Web Service ZIP File

1. Open <http://access.redhat.com> in your web browser.
2. In the right upper corner, click the **Log in** button.
3. Enter your login information and click **Log in**.
4. Hover over the **Downloads** option in the menu bar at the top of the page and click **Downloads** under **JBoss Enterprise Middleware** in the displayed menu.
5. In the **Product** drop-down menu, click **Web Server**.
6. On the **Releases** tab, click the **JBoss EWS** link for Windows with the correct architecture and version.
7. On the Software Details page, click the **Download** button next to the File drop-down box.



Note

Make sure the file is intended for your operating system and architecture.

8. Confirm the file download.

² <https://access.redhat.com>

Procedure 4.3. Extracting the ZIP File

1. Right-click the file and select **Extract All...**
2. Click **Next**.
3. When prompted for the extract location, click **Browse** and navigate to **C:\Program Files**.
4. Click **Finish**.



Note

Files are extracted in **C:\Program Files\Red Hat\Enterprise Web Server**. If you wish to change the httpd configuration (for example, add `mod_cluster`), edit the `httpd.conf` file in the **C:\Program Files\Red Hat\Enterprise Web Server\etc\httpd\conf** folder.

4.3. Configuring the Environment

Before running JBoss Enterprise Web Server, set the environment variables and run the post-installation script.

Procedure 4.4. Setting Environment Variables

1. Log in to an account with local administrator permissions.
2. Go to **Control Panel** → **System**.
3. Click on the **Advanced** tab.
4. Click the **Environment Variables** button.
5. Click the **New** for System Variables.
6. Enter the required **JAVA_HOME**, **TMP** and **TEMP** name-value pairs.
7. You will also need to add `<EWS_HOME>\bin` to the **PATH** environment variable of the user the service runs under. This is **SYSTEM** by default. Without this, the SSL Connector will fail to work.

Procedure 4.5. Running the Post-Installation Script

1. Run the **Command Prompt** as administrator even if you are logged in as the administrator:
 - a. Click the **Start** button.
 - b. Go to **All Programs** → **Accessories**.
 - c. Right-click the **Command Prompt** and select **Run as administrator**.



Running Command Prompt in Windows Server 2003

Windows Server 2003 does not have the *User Account Control* concept, therefore you only need to run the **Command Prompt** (`cmd.exe`).

2. In the **Command Prompt**, type the following to change to the **etc** folder of your Enterprise Web Server installation:

```
C:\> cd /D "C:\Program Files\Red Hat\Enterprise Web Server\etc"
```

3. Issue the command **call postinstall.bat** to run the post-installation script.

The script creates the required symbolic links (Junction Points) for temporary logging and configuration directories.

Procedure 4.6. Installing Tomcat

Follow this procedure to install Tomcat as a service:

- Run the **Command Prompt** as administrator:
 - a. Click the **Start** button.
 - b. Go to **All Programs** → **Accessories**.
 - c. Right-click the **Command Prompt** and select **Run as administrator**.



32-bit vs 64-bit Command Prompt

On 64-bit Windows, you need to use the 64-bit version of the **Command Prompt** (**cmd.exe**). Running the **cmd** command from a 32-bit application launches a 32-bit Command Prompt instance. This causes the service setup script to fail with a File Not Found warning.

- d. Change to the **sbin** folder in the JBoss Enterprise Web Server installation folder:

```
C:\> cd /D "C:\Program Files\Red Hat\Enterprise Web Server\sbin"
```

- e. In the Command Prompt, run the following command with the required Tomcat version (5 or 6) to install Tomcat:

```
call service<VERSION>.bat install
```

You can check the service parameters by typing **tomcat<VERSION>w.exe**.

Procedure 4.7. Installing HTTP

Follow this procedure to install Tomcat as a service:

1. Run the **Command Prompt** as administrator:
 - a. Click the **Start** button.
 - b. Go to **All Programs** → **Accessories**.
 - c. Right-click the **Command Prompt** and select **Run as administrator**.
2. Change to the **bin** folder of your new installation:

```
C:\> cd /D "C:\Program Files\Red Hat\Enterprise Web Server\bin"
```

3. In the Command Prompt, run the **install** command:

```
httpd -k install
```

A Firewall security dialog prompt demanding networking access for the Apache HTTP Server service appears. Click **Allow** if you wish to access this service from the network.



Note

By default, Apache HTTP Server is configured to listen on port 80. If you have Microsoft IIS installed, ensure the following:

- **World Wide Web...** service is stopped and the Startup Type is set to **Manual**
- **World Wide Web...** service is configured to listen on a different port

Alternatively, you can edit the **httpd.conf** file before installing the service and change the **Listen** directive to a port that does not collide with the Internet Information Service listening ports.

Procedure 4.8. Configuring log4j

Follow this procedure to use **log4j** logging in Tomcat:

1. Change to **C:\Program Files\Red Hat\Enterprise Web Server\extras**
2. Copy the respective log4j files to the Tomcat folder:

If using Tomcat 6, run the following commands:

```
copy log4j.jar log4j.properties ..\tomcat6\lib
```

```
copy C:\Program Files\Red Hat\Enterprise Web Server\extras\tomcat-juli-adapters.jar  
C:\Program Files\Red Hat\Enterprise Web Server\tomcat6\lib
```

If using Tomcat 5, run the following commands:

```
copy log4j.properties ..\tomcat5\common\classes
```

```
copy log4j.jar ..\tomcat5\common\lib
```

Procedure 4.9. Removing SSL

JBoss Enterprise Web Server supports SSL by default, but it may be removed.

Follow this procedure to remove or again add SSL:

1. Go to **C:\Program Files\Red Hat\Enterprise Web Server\etc\httpd\conf.d**
2. Rename the SSL configuration file:

To remove SSL, rename **ssl.conf** to **ssl.conf.disabled**.

To re-add SSL rename **ssl.conf.disabled** to **ssl.conf**.

Procedure 4.10. Configuring mod_jk

To configure the HTTP Server to use mod_jk, the following needs to be defined:

- available workers (JBoss instances) in the **workers.properties** file
- the mod_jk configuration file **mod_jk.conf**

Follow this procedure to configure the HTTP Server to use mod_jk as its load balancer:

1. In **C:\Program Files\Red Hat\Enterprise Web Server\etc\httpd\conf** create **workers.properties** (refer to <http://tomcat.apache.org/connectors-doc/reference/apache.html>).
2. In **C:\Program Files\Red Hat\Enterprise Web Server\etc\httpd\conf.d**, create **mod_jk.conf** (refer to <http://tomcat.apache.org/connectors-doc/reference/apache.html>)



Configuration Samples

You can also use the sample files **mod_jk.conf.sample** and **workers.properties.sample** available in **C:\Program Files\Red Hat\Enterprise Web Server\etc\httpd\conf**: modify their content as needed and rename them to **mod_jk.conf** and **workers.properties**.

Procedure 4.11. Configuring mod_cluster

Follow this procedure to configure the HTTP Server to use mod_cluster as its load balancer:

1. In the **C:\Program Files\Red Hat\Enterprise Web Server\etc\httpd\conf\httpd.conf** file, add the hash (#) sign at the beginning of the following line to disable **mod_proxy_balancer.so**:

```
LoadModule proxy_balancer_module /modules/mod_proxy_balancer.so
```

This module is incompatible with the JBoss HTTP Connector.

2. Configure the server to load the JBoss HTTP Connector modules:
 - a. Create the file **C:\Program Files\Red Hat\Enterprise Web Server\etc\httpd\conf\JBoss_HTTP.conf**.
 - b. Add the following lines to the **JBoss_HTTP.conf** file:

```
LoadModule slotmem_module /modules/mod_slotmem.so
LoadModule manager_module /modules/mod_manager.so
LoadModule proxy_cluster_module /modules/mod_proxy_cluster.so
LoadModule advertise_module /modules/mod_advertise.so
```

Procedure 4.12. Setting Security

Follow this procedure to ensure that the account used to run the services has full control over the **C:\Program Files\Red Hat** folder and all of its subfolders:

1. Right-click the **C:\Program Files\Red Hat** folder and click **Properties**.

2. Select the **Security** tab.
3. Click the **Add..** button and type in **LOCAL SERVICE**.
4. Select the **Full Control** check box for the new **LOCAL SERVICE** account.
5. Click the **Advanced** button.
6. Inside the *Advanced Security Settings for Red Hat* dialog, select the **Replace permissions on all child objects...** check box and click the **OK** button.

This procedure is intended for Windows 2003. The equivalent procedure in Windows Server 2008 and Windows Server 2008R2 can slightly differ.

4.4. Running Enterprise Web Server

You can run the httpd and tomcat services from the Command Prompt or with the Computer Management tool.

Procedure 4.13. Running Enterprise Web Server from the Command Prompt

Follow this procedure to run the httpd and tomcat services from the Command Prompt:

1. Run the **Command Prompt** as administrator:
 - a. Click the **Start** button.
 - b. Go to **All Programs** → **Accessories**.
 - c. Right-click the **Command Prompt** and select **Run as administrator**.
2. On the Command Prompt, issue the following command to run the tomcat service:

```
net start tomcat<VERSION>
```

3. On the Command Prompt, run the following command to run the httpd service:

```
net start httpd
```

Procedure 4.14. Running Enterprise Web Server from the Computer Management Tool

You can run the httpd and tomcat services from the Command Prompt or with the Computer Management tool.

1. Run the **Command Prompt** as administrator:
 - a. Click the **Start** button.
 - b. Go to **All Programs** → **Accessories**.
 - c. Right-click the **Command Prompt** and select **Run as administrator**.
2. Go to **Start** → **Administrative Tools** → **Services**
3. In the Services dialog, right-click the name of the service (**httpd** or **tomcat**) and click **Start** or **Stop**.

**Note**

Some thirdparty applications add libraries to the system directory in Windows. These take precedence over Tomcat when DLL libraries are looked up. This means that if they have the same name as the those used by Tomcat native libraries, they are loaded instead of the libraries distributed with JBoss Enterprise Web Server. Tomcat therefore does not start, and does not long any error messages about it in Windows Event Log, or Tomcat log files. Errors can only be seen by using `catalina.bat run`.

If this behaviour occurs, check the content of the `C:\windows\System32\` directory and PATH directories, and check for DLLs library conflicting with those shipped with Enterprise Web Server. In particular, look for `libeay32.dll`, `ssleay32.dll`, and `libssl32.dll`.

4.5. Stopping Enterprise Web Server

You can run the httpd and tomcat services from the Command Prompt or with the Computer Management tool.

Procedure 4.15. Stopping Enterprise Web Server from the Command Prompt

Follow this procedure to stop the httpd and tomcat services from the Command Prompt:

1. Run the **Command Prompt** as administrator:
 - a. Click the **Start** button.
 - b. Go to **All Programs** → **Accessories**.
 - c. Right-click the **Command Prompt** and select **Run as administrator**.
2. On the Command Prompt, issue the following command to run the tomcat service:

```
net stop tomcat<VERSION>
```

3. On the Command Prompt, run the following command to run the httpd service:

```
net stop httpd
```

Procedure 4.16. Stopping Enterprise Web Server from the Computer Management Tool

Follow this procedure to run the httpd and tomcat services with the Computer Management tool:

1. Run the **Command Prompt** as administrator:
 - a. Click the **Start** button.
 - b. Go to **All Programs** → **Accessories**.
 - c. Right-click the **Command Prompt** and select **Run as administrator**.
2. Go to **Start** → **Administrative Tools** → **Services**
3. In the Services dialog, right-click the name of the service (**httpd** or **tomcat**) and click **Stop**.

4.6. Folder Structure

The following is the default folder layout after a successful installation:

```
|-- bin
|-- doc
|-- etc
|   |-- httpd
|   |-- postinstall.bat
|   |-- ssl
|   `-- sysconfig
|-- include
|-- lib[64]
|-- sbin
|-- share
|   |-- apache-tomcat-5.VERSION
|   |-- apache-tomcat-6.VERSION
|   |-- java
|   |-- tomcat5
|   `-- tomcat6
`-- var
    |-- cache
    |-- log
    |-- run
    `-- www
```

The Windows folder structure follows Red Hat Enterprise Linux conventions:

- The **bin** and **sbin** directories contain the executable files for Apache HTTP Server and Tomcat.
- The **etc** folder contains configuration files for Apache HTTP Server and the post-installation script.
- The **share** folder contains Apache Tomcat 5 and 6 installations and shared Java JARs.
- The **var** contains four sub-directories:

cache

The **cache** contains resources cached by the server..

log

This **log** folder contains the log files for HTTP Server and Tomcat.

www

Apache HTTP Server looks in this folder for web content in the **default** configuration.

run

The **run** folder contains PID files.

Installing Enterprise Web Server on Solaris

On Solaris, JBoss Enterprise Web Server can be installed in two ways:

- from a ZIP file
- from Solaris packages

Both installation ZIP files and Solaris packages are available at [Red Hat Customer Portal](#)¹; however, their installation differs. JBoss Enterprise Web Server is run in the same way regardless of whether it was installed from a ZIP or Solaris package.

5.1. Prerequisites

Before installing JBoss Enterprise Web Server on Solaris, confirm that a supported Java version is installed (refer to [Procedure 5.1, “Installing Java”](#)).

Procedure 5.1. Installing Java

1. Go to the [Oracle homepage](#)².
2. Download JDK 1.6.0 from the web site (<http://www.oracle.com/technetwork/java/javase/downloads/index.html>).
3. Go to the download location.
4. Run the Java installation file.
5. Go to **usr**.
6. Issue the following command to display the current Java symbolic link:

```
ls -lad java
```

7. Remove the link:

```
rm java
```

8. Create a java symbolic link to the JDK 1.6.0:

```
ln -sf /usr/jdk/<JDK>
```

5.2. Downloading Installation Files

Both installation resources, ZIP and Solaris packages, are available at the [Red Hat Customer Portal](#)³.

Procedure 5.2. Downloading Solaris Packages or ZIP Files

Follow this procedure to download ZIP files or Solaris package with JBoss Enterprise Web Server:

¹ <https://access.redhat.com>

³ <https://access.redhat.com>

1. Open <https://access.redhat.com> in your web browser.
2. In the right upper corner, click the **Log in** button.
3. Enter your login information and click **Log in**.
4. Hover over the **Downloads** option in the menu bar at the top of the page and click **Downloads** under **JBoss Enterprise Middleware** in the displayed menu.
5. In the **Product** drop-down box column of the displayed page, select **Web Server** wait for the page to refresh.
6. On the **Releases** tab, click the **JBoss EWS** link for your operating system and architecture.
7. On the Software Details page, click the **Download** button next to the File drop-down box.



Note

Make sure the file is intended for your operating system and architecture.

8. Confirm the file download.

5.3. Extracting and Installing Enterprise Web Server

Extraction and installation of Enterprise Web Server differs for the ZIP and Solaris packages. Depending on the resource you have downloaded proceed to either [Procedure 5.3, “Extracting and Installing EWS from a ZIP File”](#) or to [Procedure 5.4, “Extracting and Installing EWS from Solaris Packages”](#).

Procedure 5.3. Extracting and Installing EWS from a ZIP File

Follow this procedure to install JBoss Enterprise Web Server from a ZIP file:

1. At the shell prompt, become the root user.
2. Change to the directory with the ZIP file.
3. Run the following command to extract the package to **opt**:

```
# unzip <FILE_NAME> -d /opt
```

Substitute the `<FILE_NAME>` with the name of the downloaded ZIP file.

4. Issue the following commands to run the post-installation script:

```
cd /opt/redhat/ews/etc  
sh .postinstall
```

The post-installation process creates the tomcat and apache users and groups, and directories:

- The users and groups apache with an ID of 48 and tomcat with an ID of 91. The apache runs **httpd**, while the tomcat user runs **tomcat**.
- Additional directories the tomcat group has write permissions to:
 - **/var/logs/tomcat5**

- `/var/logs/tomcat6`

Procedure 5.4. Extracting and Installing EWS from Solaris Packages

Before starting the installation make sure you have root access to the system and that you have downloaded the Solaris package intended for your Solaris version and architecture (refer to [Section 5.2, "Downloading Installation Files"](#)).

When installing from a Solaris package, the system installs JBoss Enterprise Web Server automatically in the `/opt/redhat/ews` directory. As a part of the installation, the package installer executes custom scripts, which set up the apache and tomcat user accounts.

Follow this procedure to install JBoss Enterprise Web Server from Solaris packages:

1. At the shell prompt, become the root user.
2. Run the following `gunzip` and `pkgadd` commands to decompress and install the package:

```
gunzip RHATews-1.0.*
```

```
pkgadd -d RHATews-1.0.*
```

The command returns an output similar to the following:

```
The following packages are available:
1  RHATews      JBoss Enterprise Web Server (i386) 1.0.0,REV=2.e15

Select package(s) you wish to process (or 'all' to process all packages). (default: all)
[?,??,q]:
```

3. Type **1** or press **enter** to confirm the installation. The following information appears:

```
JBoss Enterprise Web Server(i386) 1.0.0,REV=2.GA

END USER LICENSE AGREEMENT
JBOSS(r) ENTERPRISE MIDDLEWARE(tm)

The end user license agreement ("EULA") governs the use of the various software modules
that collectively comprise JBoss Enterprise Middleware
and any related updates, source code, appearance, structure and organization,
regardless of the delivery mechanism.

The JBoss Enterprise Middleware EULA can be found here:
http://www.redhat.com/licenses/jboss_eula.html

## Executing checkinstall script.
Using </opt> as the package base directory.
## Processing package information.
## Processing system information.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
## Checking for setuid/setgid programs.

This package contains scripts which will be executed with super-user
permission during the process of installing this package.
```

```
Do you want to continue with the installation of <RHATews> [y,n,?]
```

4. Type **y** and press **enter** when prompted to confirm the installation. Installation starts and the following appears:

```
## Executing postinstall script.
Apache group (id=48) already exists. +++ or created if not
Apache user (id=48) already exists.
Generating private RSA key ... OK
Generating new (+++ user hostname displayed here) certificate ... OK
Tomcat group (id=91) already exists.
Tomcat user (id=91) already exists.
```

```
-----
NOTICE
-----
```

```
JAVA_HOME environment variable is not set.
Either set the JAVA_HOME or edit the configuration
scripts inside '/opt/redhat/ews/etc/sysconfig' directory
and set the JAVA_HOME to the installed JDK location.
```

```
Installation of <RHATews> was successful.
```

5. Issue the following command to display the package information:

```
pkginfo -l RHATews
```



Note

During installation, the system generates a notice that the `JAVA_HOME` environment variable is not set. You need to set this variable specifically for the Tomcat you wish to run (refer to [Procedure 5.5, “Configuring Tomcat”](#) section).

6. Run the following command with the respective Tomcat version to execute the `catalina.sh` script:

```
/opt/redhat/ews/share/tomcat<VERSION>/bin/catalina.sh
```

The `catalina.sh` script detects the running JVM version and adds the `/opt/redhat/ews/lib` path (`/opt/redhat/ews/lib64` for 64 bit JVM) to the library path.

5.4. Configuring the Environment

After you have installed JBoss Enterprise Web Server, set the `JAVA_HOME` for your Tomcat (refer to [Procedure 5.5, “Configuring Tomcat”](#)).

Optionally, you can also:

- remove SSL (refer to [Procedure 5.6, “Removing SSL”](#));

- add log4j logging (refer to [Procedure 5.7, “Configuring log4j”](#));
- enable mod_jk or mod_cluster (refer to [Procedure 5.8, “Configuring mod_jk”](#) and [Procedure 5.9, “Configuring mod_cluster”](#)).

Procedure 5.5. Configuring Tomcat

Before starting Tomcat, set its JAVA_HOME variable to point to a supported JDK (1.6.0):

1. Locate the **tomcat** configuration file:
 - For Tomcat 5: **/opt/redhat/ews/etc/sysconfig/tomcat5**
 - For Tomcat 6: **/opt/redhat/ews/etc/sysconfig/tomcat6**
2. Open the configuration file and delete the hash (#) sign at the beginning of the following line:

```
# JAVA_HOME="/usr/java"
```

Procedure 5.6. Removing SSL

JBoss Enterprise Web Server by default supports SSL. However, you may remove the SSL support.

Follow this procedure to remove or add SSL:

1. Go to **/opt/redhat/ews/etc/httpd/conf.d**
2. Rename the SSL configuration file:

To remove SSL, rename **ssl.conf** to **ssl.conf.disabled**.

To re-enable the SSL, rename **ssl.conf.disabled** to **ssl.conf**.

Procedure 5.7. Configuring log4j

Follow this procedure to add **log4j** logging to Tomcat:

1. Go to the **/opt/redhat/ews/share/extras** directory.
2. Copy the **log4j.jar** and **log4j.properties** files to the **lib** directory of the Tomcat directory:

If using Tomcat 6, run the following commands:

```
extras]# cp log4j.jar log4j.properties ../tomcat6/lib
```

```
extras] # cp tomcat-juli-adapters.jar /opt/redhat/ews/share/tomcat6/lib
```

If using Tomcat 5, run the following commands:

```
extras]# cp log4j.properties ../tomcat5/common/classes
```

```
extras]# cp log4j.jar ../tomcat5/common/lib
```

Procedure 5.8. Configuring mod_jk

To configure the HTTP Server to use mod_jk, define the following:

- available workers (JBoss instances) in the **workers.properties** file
- the mod_jk configuration file **mod_jk.conf**

Follow this procedure to configure the HTTP Server to use mod_jk as its load balancer:

1. Create a workers.properties file in `/opt/rehat/ews/etc/httpd/conf/httpd/conf/` (refer to <http://tomcat.apache.org/connectors-doc/reference/apache.html>).
2. Create a mod_jk.conf in `/opt/rehat/ews/etc/httpd/conf/httpd/conf.d/`, (refer to <http://tomcat.apache.org/connectors-doc/reference/apache.html>).



Configuration Samples

You can also copy and modify the sample files `mod_jk.conf.sample` and `workers.properties.sample` available in `/opt/rehat/ews/etc/httpd/conf/httpd/conf/`: modify their content as needed and rename them to `mod_jk.conf` and `workers.properties`.

Procedure 5.9. Configuring mod_cluster

1. From [Red Hat Customer Portal](#)⁴, download the jboss-ep-native ZIP file for your operating system and architecture.
2. Unzip the downloaded file.
3. In the location, where you have extracted the file, change to `jboss-ep-<VERSION> /native/lib[64]/httpd/modules`
4. Run the following command to copy mod_cluster modules to `/opt/redhat/ews/lib[64]/httpd/modules`:

```
modules]# cp mod_advertise.so mod_manager.so mod_proxy_cluster.so mod_slotmem.so -t /usr/lib[64]/httpd/modules
```

5. In `/opt/redhat/ews/httpd/conf/httpd.conf` add a hash (#) sign at the beginning of the following line to disable the mod_proxy_balancer module:

```
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
```

This module is incompatible with the JBoss HTTP Connector.

6. Configure the server to load the JBoss HTTP Connector modules:
 - a. Create the file `/opt/redhat/ews/httpd/conf.d/JBoss_HTTP.conf`.
 - b. Add the following lines to the file `$EWS_HOME/httpd/conf.d/JBoss_HTTP.conf`:

```
LoadModule slotmem_module $EWS_HOME/httpd/modules/mod_slotmem.so
LoadModule manager_module $EWS_HOME/httpd/modules/mod_manager.so
LoadModule proxy_cluster_module $EWS_HOME/httpd/modules/mod_proxy_cluster.so
LoadModule advertise_module $EWS_HOME/httpd/modules/mod_advertise.so
```

5.5. Running Enterprise Web Server

After the installation is complete, you can run the required Enterprise Web Server components.

5.5.1. Running Tomcat

Tomcat can be run either in the foreground as a program or in the background as a service.



Supported Method of Running Tomcat

We strongly recommend to run tomcat as a service (refer to [Section 5.5.1.2, “Running Tomcat as a Service”](#)). Note that this is the only supported running method.

5.5.1.1. Running Tomcat as a Program

If you want to run Tomcat as a program, use either the `tomcat[5,6]` script in the `sbin` directory or the `catalina` script in the `lib` directory:

- To start Tomcat with the `tomcat` script, run the following command as the root user with the respective Tomcat version (**5** or **6**):

```
/opt/redhat/ews/sbin/tomcat<VERSION> start
```

The system reads the configuration file and changes to the user and group of the tomcat user account (that is, Tomcat is always run under the tomcat user). By default, `tomcat-native` is automatically added to the `LD_LIBRARY_PATH` and `-Djava.library.path` during Tomcat startup.

- To start Tomcat with the `catalina.sh` script, run the following command with the respective Tomcat version (**5** or **6**):

```
/opt/redhat/ews/share/tomcat<VERSION>/bin/catalina.sh
```

The `catalina.sh` script detects the running JVM version and adds the `/opt/redhat/ews/lib/` path (`/opt/redhat/ews/lib64` for 64 bit JVM) to the library path.

5.5.1.2. Running Tomcat as a Service

To run Tomcat as a service you can use either the Tomcat script located in the `init.d` or you can create and use your own system daemon script with `jsvc` or `tanukiwrapper`.

To run Tomcat as a service, create symbolic links to the provided Tomcat scripts using the following command:

```
ln -s /opt/redhat/ews/etc/init.d/tomcat<VERSION> /etc/rc3.d/S70tomcat<VERSION>
ln -s /opt/redhat/ews/etc/init.d/tomcat<VERSION> /etc/rc5.d/K20tomcat<VERSION>
```

Define the `init` level numbers (for information on how to define `init` level numbers consult your Solaris documentation). The preferred method of service management is through the **Solaris Service Management Facility**. In order to read more about this facility, consult the **smf(5)** manual.

5.5.2. Running HTTP Server

The HTTP Server can be run with the provided `apachectl` script, which is located inside the `/opt/redhat/ews/sbin` directory. The `apachectl` script uses the configuration file `/opt/redhat/ews/`

`etc/sysconfig/httpd`. In this file you can edit and select startup parameters, such as the running of `mpm`.

Apache `httpd` configuration files are located inside the `/opt/redhat/ews/etc/httpd` directory. By default all supported modules are loaded and enabled (refer to [Appendix A, Supported HTTP Modules](#)). To disable some of the modules, comment out the respective modules in the `httpd.conf` file (for module description refer to <http://httpd.apache.org/docs/2.2/mod/>).

Procedure 5.10. Running HTTP Server

Follow this procedure to start the HTTP Server:

1. Run the command line as the root user.
2. Run the following command to start the HTTP Server:

```
sbin/apachectl start
```

If you want your server to continue running automatically after a system reboot, add a call to the `apachectl` script to your system startup files.



Note

For complete documentation on how to start and stop the HTTP Server, refer to the files, `docs/httpd/invoking.html` and `docs/httpd/stopping.html` respectively.

5.6. Stopping Enterprise Web Server

Procedure 5.11. Stopping Tomcat Running a Service

Follow this procedure to stop tomcat, which is running as a service:

1. At the shell prompt, become the root user.
2. Execute the following command with the respective Tomcat version (**5** or **6**) to stop Tomcat started from with the Tomcat script in the `bin` directory:

```
sbin/tomcat<VERSION> stop
```

Procedure 5.12. Stopping Tomcat Running a Program

Follow this procedure to stop tomcat, which is running as a program:

1. At the shell prompt, change to `/opt/redhat/ews/share/tomcat<VERSION>/bin`.
2. Execute the following command with the respective Tomcat version (**5** or **6**) to stop Tomcat started from with the Tomcat script in the `bin` directory:

```
sh shutdown.sh
```

Procedure 5.13. Stopping HTTP Server

Follow this procedure to start the HTTP Server:

1. Run the command line as the root user.

2. Run the following command to stop the HTTP Server:

```
sbin/apachectl stop
```

5.7. Uninstalling the Package in Solaris

Follow this procedure to uninstall a Solaris package installation of JBoss Enterprise Web Server:

1. Issue the following command to uninstall the package:

```
pkgrm RHATews
```

2. Remove the following directories:

- **/opt/redhat/ews**
- **/var/run/tomcat5**
- **/var/run/tomcat6**
- **/var/cache/tomcat5**
- **/var/cache/tomcat6**
- **/var/log/httpd**
- **/var/cache/mod_ssl**
- **/var/cache/mod_proxy**

5.8. Solaris Directory Structure

The following is the default directory layout after a successful installation:

```
.
|- /opt/redhat/ews/
    |- bin
    |- doc
    |- etc
    |- lib
    |- lib64 (present only for x86_64 systems platforms)
    |- include
    |- README
    |- man
    |- share
        |- tomcat5
        |- tomcat6
        |- extras
        |- java
    |- sbin
    |- var
```

The Solaris directory structure is different to the Red Hat Enterprise Linux structure. This is because:

- Native Solaris packages require the absolute location
- Solaris (and other non-Red Hat Enterprise Linux platforms) must ship and build all the dependent components such as **openldap**, **openssl**, **db4** and **cyrus-sasl**.

Part II. Upgrade

The upgrade procedure of JBoss Enterprise Web Server differs for individual operating systems. Refer to the upgrade chapter for your operating system to upgrade JBoss Enterprise Web Server :

- for Red Hat Enterprise Linux [Chapter 6, Upgrading Enterprise Web Server on Red Hat Enterprise Linux](#)
 - for Windows [Chapter 7, Upgrading Enterprise Web Server on Windows](#)
 - for Solaris [Chapter 8, Upgrading Enterprise Web Server on Solaris](#)
-

Upgrading Enterprise Web Server on Red Hat Enterprise Linux

Follow the upgrade procedure for your installation type to upgrade a previous version of JBoss Enterprise Web Server on Red Hat Enterprise Linux (refer to [Section 6.1, “Upgrading ZIP Installation”](#) or [Section 6.2, “Upgrading RPM Installation”](#)).

6.1. Upgrading ZIP Installation

Follow this procedure to upgrade JBoss Enterprise Web Server installed from a ZIP file:

1. Back up your applications and custom settings.
2. Install the new JBoss Enterprise Web Server version (refer to [Section 3.2, “Installing Enterprise Web Server from a ZIP File”](#)).
3. Copy the applications and custom settings to the directory with the new JBoss Enterprise Web Server installation.
4. Make sure all applications work correctly before removing their backups.

6.2. Upgrading RPM Installation

If you are using a previous version of JBoss Enterprise Web Server installed from RPM packages, upgrade your system to upgrade to the new JBoss Enterprise Web Server version (refer to [Procedure 6.1, “Upgrading JBoss Enterprise Web Server on Red Hat Enterprise Linux 5”](#) or [Procedure 6.2, “Upgrading JBoss Enterprise Web Server on Red Hat Enterprise Linux 4”](#)).



Software Update

You can upgrade to the new version also using System Update.

Procedure 6.1. Upgrading JBoss Enterprise Web Server on Red Hat Enterprise Linux 5

Do the following to upgrade your JBoss Enterprise Web Server RPM installation on Red Hat Enterprise Linux 5:

1. At the shell prompt, become the root user.
2. Run the following command to remove the classpathx-jaf package:

```
# yum remove classpathx-jaf
```

3. Run the upgrade command:

```
# yum upgrade
```

Procedure 6.2. Upgrading JBoss Enterprise Web Server on Red Hat Enterprise Linux 4

Upgrade your system to upgrade your JBoss Enterprise Web Server RPM installation on Red Hat Enterprise Linux 4:

1. At the shell prompt, become the root user.
2. Run the following command:

```
# up2date -u
```

Upgrading Enterprise Web Server on Windows

Follow this procedure to upgrade JBoss Enterprise Web Server on Windows:

1. Back up your applications and custom settings.
2. Install the new JBoss Enterprise Web Server version (refer to [Chapter 4, Installing Enterprise Web Server on Windows](#)).
3. Copy the applications and custom settings to the directory with the new JBoss Enterprise Web Server installation (**C:\Program Files\Red Hat\Enterprise Web Server**).
4. Make sure all applications work correctly before removing the backup.

Upgrading Enterprise Web Server on Solaris

Follow this procedure to upgrade JBoss Enterprise Web Server on Solaris:

1. Back up your applications and custom settings.
2. Install the new JBoss Enterprise Web Server version (refer to [Chapter 5, Installing Enterprise Web Server on Solaris](#)).
3. Copy the applications and custom settings to the directory with the new JBoss Enterprise Web Server installation (`/opt/redhat/ews/`).
4. Make sure all the applications work correctly before removing the backup.

Hibernate on Enterprise Web Server

Hibernate is an object-relational mapping framework. It is delivered in a separate package and the same package can be used on all supported platforms.

Hibernate can be used in the same way it is used in a regular Tomcat installation; that is, the Hibernate JAR files can be added into your deployment WAR file. Tomcat provides a connection pooling mechanism by default, and you should consider taking advantage of it (the pooling mechanism is defined in the `context.xml`). However, `persistence.xml` and `web.xml` are required. The example below shows a setting with tomcat connection pooling mechanism.

- `/META-INF/context.xml` defines the connection pools Tomcat should create.

Example 9.1. context.xml

```
<Context>
  <Resource
    name="jdbc/DsWebAppDB"
    auth="Container"
    type="javax.sql.DataSource"
    username="sa"
    password=""
    driverClassName="org.h2.Driver"
    url="jdbc:h2:mem:target/test/db/h2/hibernate"
    maxActive="8"
    maxIdle="4"/>
</Context>
```

- `/WEB-INF/classes/META-INF/persistence.xml` is a JPA configuration file. It defines how the application configures Hibernate to consume connections from the Tomcat pool. If you are using Hibernate API directly, use a similar configuration in `hibernate.cfg.xml`.

Example 9.2. persistence.xml

```
<persistence version="1.0"
  xmlns="http://java.sun.com/xml/ns/persistence" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/persistence http://java.sun.com/xml/ns/
persistence/persistence_1_0.xsd">

  <persistence-unit name="dswebapp">
    <provider>org.hibernate.ejb.HibernatePersistence</provider>
    <properties>
      <property name="hibernate.dialect" value="org.hibernate.dialect.H2Dialect" />
      <property name="hibernate.connection.datasource" value="java:comp/env/jdbc/
DsWebAppDB"/>
    </properties>
  </persistence-unit>
</persistence>
```

- `/WEB-INF/web.xml` is a regular web application deployment file, which tells Tomcat which datasource it consumes. In [Example 9.3, "web.xml"](#) the datasource is `jdbc/DsWebAppDB`.

Example 9.3. web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
  java.sun.com/xml/ns/javaee
  http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">

  <resource-env-ref>
    <resource-env-ref-name>jdbc/DsWebAppDB</resource-env-ref-name>
    <resource-env-ref-type>javax.sql.DataSource</resource-env-ref-type>
  </resource-env-ref>
</web-app>
```

For details, refer to the Hibernate documentation for JBoss Enterprise Web Server .

Monitoring Enterprise Web Server with JBoss Operating Network

To allow JBoss Operating Network to monitor JBoss Enterprise Web Server, you need to configure the server to allow the JBoss ON discovery and give JBoss ON the needed access.

Procedure 10.1. Configuring User Permissions on a Unix Operating System



Note

If you are using Windows, skip these steps and proceed to [Procedure 10.2, “Configuring Monitoring”](#).

RHQ/JON agent needs read and write permission to the **httpd** and Tomcat directories.

Follow this procedure to assign the permissions:

1. At the shell prompt, become the root user.
2. Run the following command to add the user, under which JON Agent runs to the tomcat and apache user groups:

```
# useradd -G tomcat,apache <RHQ_AGENT_USER>
```

Procedure 10.2. Configuring Monitoring

JBoss Enterprise Web Servers are auto-discovered on Linux and Unix platforms. However, you need to configure the instance's JMX to allow proper handling of authentication and accurate Tomcat monitoring.

Follow this procedure to set up JMX to handle authentication:

1. Open the **startup** file of the respective JBoss Enterprise Web Server instance for editing:

on Red Hat Enterprise Linux installed from a *ZIP* file open **`$EWS_HOME/tomcat<VERSION>/bin/startup.sh`**

on Red Hat Enterprise Linux installed from a *RPM* files open **`/usr/sbin/tomcat<VERSION>`**

on Solaris open **`/opt/redhat/ews/tomcat<VERSION>/bin/startup.sh`**

on Windows open **`C:\Program Files\Red Hat\Enterprise Web Server\sbin\tomcat<VERSION>.sh`**

2. Define an available port for JMX monitoring. Make sure it is not blocked by any firewall. To do so, to the `JAVA_OPTS` variable in the startup file, add the following line and substitute `PORT_NUMBER` with the number of the port dedicated to monitoring:

on Red Hat Enterprise Linux and Solaris:

```
JAVA_OPTS="${JAVA_OPTS} -Dcom.sun.management.jmxremote.port=PORT_NUMBER"
```

on Windows:

```
set "JAVA_OPTS=%JAVA_OPTS% -Dcom.sun.management.jmxremote.port=PORT_NUMBER"
```

3. If you want to disable authentication and SSL for development purposes, add the following lines to the JAVA_OPTS variable in the startup file:

```
JAVA_OPTS="${JAVA_OPTS} -Dcom.sun.management.jmxremote.ssl=false"  
JAVA_OPTS="${JAVA_OPTS} -Dcom.sun.management.jmxremote.authenticate=false"
```



Important

In production environments, add the following lines to the JAVA_OPTS variable in the startup file to secure JMX with SSL and restrict the access with a firewall:

on Red Hat Enterprise Linux and Solaris

```
JAVA_OPTS="${JAVA_OPTS} -Dcom.sun.management.jmxremote.access.file=$EWS_HOME/  
jmxremote.access"  
JAVA_OPTS="${JAVA_OPTS} -Dcom.sun.management.jmxremote.password.file=$EWS_HOME/  
jmxremote.password"
```

on Windows:

```
set "JAVA_OPTS=%JAVA_OPTS% -Dcom.sun.management.jmxremote.access.file=c:\Program  
Files\Red Hat\Enterprise Web Server\jmxremote.access"  
set "JAVA_OPTS=%JAVA_OPTS% -Dcom.sun.management.jmxremote.password.file=c:  
\Program Files\Red Hat\Enterprise Web Server\jmxremote.password"
```



Path to the HTTP Server

The JON httpd plugin needs a specific directory to monitor the HTTP Server:

- on Solaris and Red Hat Enterprise Linux: **/etc/httpd**
- on Windows: **C:\apache**

If the respective directory does not exist, create it before launching the JON agent.

For further information about JBoss Operating Network refer to the [Discovery chapter of the Basic Administration Guide](#)¹.

Appendix A. Supported HTTP Modules

- mod_filter.so
- mod_filter.so
- mod_substitute.so
- mod_version.so
- mod_usertrack.so
- mod_authn_alias.so
- mod_vhost_alias.so
- mod_headers.so
- mod_authz_host.so
- mod_dumpio.so
- mod_ssl.so
- mod_userdir.so
- mod_authn_dbd.so
- mod_env.so
- mod_ldap.so
- mod_expires.so
- mod_logio.so
- mod_authn_default.so
- mod_proxy_scgi.so
- mod_negotiation.so
- mod_disk_cache.so
- mod_authn_file.so
- mod_autoindex.so
- mod_auth_basic.so
- mod_proxy_http.so
- mod_setenvif.so
- mod_proxy.so
- mod_suexec.so
- mod_dav.so

Appendix A. Supported HTTP Modules

- mod_asis.so
- mod_alias.so
- mod_proxy_balancer.so
- mod_log_forensic.so
- mod_mime_magic.so
- mod_deflate.so
- mod_proxy_connect.so
- mod_dav_fs.so
- mod_cgi.so
- mod_dbd.so
- mod_mem_cache.so
- mod_cgid.so
- mod_auth_digest.so
- mod_authz_owner.so
- mod_include.so
- mod_proxy_ajp.so
- mod_reqtimeout.so
- mod_authn_anon.so
- mod_status.so
- mod_cern_meta.so
- mod_log_config.so
- mod_cache.so
- mod_dir.so
- mod_proxy_ftp.so
- mod_ext_filter.so
- mod_authz_user.so
- mod_mime.so
- mod_unique_id.so
- mod_authz_default.so
- mod_ident.so

-
- mod_file_cache.so
 - mod_speling.so
 - mod_actions.so
 - mod_authnz_ldap.so
 - mod_authn_dbm.so
 - mod_rewrite.so
 - mod_authz_groupfile.so
 - mod_imagemap.so
 - mod_info.so
 - mod_authz_dbm.so

Appendix B. Revision History

Revision Tue Jun 21 2011
1.0.2-50

Rebecca Newton rnewton@redhat.com

Final build for EWS 1.0.2.GA.
