For use with Red Hat JBoss middleware products.
Abstract

This book is a guide to the installation of Red Hat JBoss Core Services Apache HTTP Server.
# Table of Contents

## CHAPTER 1. INTRODUCTION

1.1. ABOUT RED HAT JBOSS CORE SERVICES ....................................................... 4
1.2. ABOUT JBOSS CORE SERVICES APACHE HTTP SERVER .............................. 4
1.3. SUPPORTED OPERATING SYSTEMS AND CONFIGURATIONS ..................... 4
1.4. INSTALLATION METHODS ........................................................................... 4
1.5. UPGRADE TO THE RED HAT JBOSS CORE SERVICES APACHE HTTP SERVER 2.4.29
   - Prerequisites ................................................................................................. 5
   - Procedure ....................................................................................................... 5
   - Additional Resources ..................................................................................... 5

## CHAPTER 2. INSTALLING JBOSS CORE SERVICES APACHE HTTP SERVER ON RED HAT ENTERPRISE LINUX

2.1. ZIP INSTALLATION ....................................................................................... 6
   - Prerequisites .................................................................................................. 6
   - Download and Extract Apache HTTP Server .................................................. 6
   - Configuring the Apache HTTP Server Installation ........................................... 6
     - Creating an Apache User .............................................................................. 7
     - Disabling/Enabling SSL Support ................................................................. 7
     - Running the Apache HTTP Server Post-Installation Script ......................... 7
   - Starting Apache HTTP Server ....................................................................... 7
   - Stopping Apache HTTP Server ..................................................................... 7
   - Running Apache HTTP Server without root access on Red Hat Enterprise Linux (ZIP installation) ................................................................. 8
2.2. RPM INSTALLATION ..................................................................................... 8
   - Using mod_jk, mod_cluster, mod_rt, and mod_bmx with RHEL 8 ................. 8
     - Installing httpd ............................................................................................ 9
     - Installing Modules ...................................................................................... 9
   - Installing the Apache HTTP Server from RPM Packages .............................. 9
   - Configuring the Apache HTTP Server Installation (RPM Installation) .......... 10
     - Removing SSL Support ............................................................................... 11
   - Starting Apache HTTP Server ...................................................................... 11
   - Stopping Apache HTTP Server ..................................................................... 11
   - Configuring Apache HTTP Server to Start at Boot .................................... 11
2.3. SELINUX POLICIES ....................................................................................... 12
   - SE Linux Policy Information ........................................................................ 12
   - SE Linux Policies for an RPM Installation .................................................. 12
   - SE Linux Policies for a ZIP Installation ...................................................... 12

## CHAPTER 3. INSTALLING JBOSS CORE SERVICES APACHE HTTP SERVER ON MICROSOFT WINDOWS

3.1. DOWNLOAD AND EXTRACT APACHE HTTP SERVER .................................. 14
3.2. CONFIGURING THE APACHE HTTP SERVER INSTALLATION ..................... 14
   - Running the Apache HTTP Server Post-Installation Script ......................... 14
   - Installing the Apache HTTP Server Service ............................................... 14
   - Configuring Folder Permissions for the Apache HTTP Server Service ........ 15
   - Disabling/Enabling SSL Support .................................................................. 15
3.3. STARTING APACHE HTTP SERVER ............................................................... 16
   - Starting Apache HTTP Server Using the Command Prompt ......................... 16
   - Starting Apache HTTP Server Using the Computer Management Tool ........ 16
3.4. STOPPING APACHE HTTP SERVER .............................................................. 16
   - Stopping Apache HTTP Server Using the Command Prompt ....................... 16
   - Stopping Apache HTTP Server Using the Computer Management Tool ....... 16

## CHAPTER 4. INSTALLING APACHE HTTP SERVER ON SOLARIS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ABOUT RED HAT JBOSS CORE SERVICES</td>
<td>4</td>
</tr>
<tr>
<td>2. ABOUT JBOSS CORE SERVICES APACHE HTTP SERVER</td>
<td>4</td>
</tr>
<tr>
<td>3. SUPPORTED OPERATING SYSTEMS AND CONFIGURATIONS</td>
<td>4</td>
</tr>
<tr>
<td>4. INSTALLATION METHODS</td>
<td>4</td>
</tr>
<tr>
<td>5. UPGRAWD TO THE RED HAT JBOSS CORE SERVICES APACHE HTTP SERVER 2.4.29</td>
<td>4</td>
</tr>
<tr>
<td>6. ZIP INSTALLATION</td>
<td>6</td>
</tr>
<tr>
<td>7. RPM INSTALLATION</td>
<td>8</td>
</tr>
<tr>
<td>8. SELINUX POLICIES</td>
<td>12</td>
</tr>
<tr>
<td>9. INSTALLING JBOSS CORE SERVICES APACHE HTTP SERVER ON MICROSOFT WINDOWS</td>
<td>14</td>
</tr>
<tr>
<td>10. INSTALLING JBOSS CORE SERVICES APACHE HTTP SERVER ON SOLARIS</td>
<td>17</td>
</tr>
</tbody>
</table>
4.1. DOWNLOAD AND EXTRACT APACHE HTTP SERVER
4.2. CONFIGURING THE APACHE HTTP SERVER INSTALLATION
   Running the Apache HTTP Server Post-Installation Script
   Disabling/Enabling SSL Support
4.3. STARTING APACHE HTTP SERVER
4.4. STOPPING APACHE HTTP SERVER
4.5. RUNNING APACHE HTTP SERVER WITHOUT ROOT ACCESS ON SOLARIS (ZIP INSTALLATION)

CHAPTER 5. ENABLING HTTP/2 FOR THE JBOSS CORE SERVICES HTTP SERVER
   Prerequisites
   Procedure
   Next Steps
   Additional Resources

CHAPTER 6. MONITORING APACHE HTTP SERVER WITH JBOSS OPERATIONS NETWORK (ON)
CHAPTER 1. INTRODUCTION

1.1. ABOUT RED HAT JBOSS CORE SERVICES

Red Hat JBoss Core Services is a set of supplementary software for Red Hat JBoss middleware products. This software, such as Apache HTTP Server, is common to multiple JBoss middleware products, and is packaged under Red Hat JBoss Core Services to allow for faster distribution of updates, and for a more consistent update experience.

1.2. ABOUT JBOSS CORE SERVICES APACHE HTTP SERVER

Apache HTTP Server is used in multiple Red Hat JBoss middleware products, and previously Apache HTTP Server was distributed with each JBoss product. Starting from the following product versions, each product will instead use the JBoss Core Services distribution of Apache HTTP Server:

- Red Hat JBoss Enterprise Application Platform (JBoss EAP) 7.0 and onwards.
- Red Hat JBoss Web Server 3.1 and onwards.

**IMPORTANT**

The Apache HTTP Server distribution included as part of Red Hat Enterprise Linux is separate from the JBoss Core Services distribution of Apache HTTP Server.

1.3. SUPPORTED OPERATING SYSTEMS AND CONFIGURATIONS

For information on supported operating systems and configurations for JBoss Core Services Apache HTTP Server, see [https://access.redhat.com/articles/2258971](https://access.redhat.com/articles/2258971).

1.4. INSTALLATION METHODS

JBoss Core Services Apache HTTP Server can be installed on supported Red Hat Enterprise Linux, Microsoft Windows, and Solaris systems using ZIP installation files available for each platform. JBoss Core Services Apache HTTP Server can also be installed on supported Red Hat Enterprise Linux systems using RPM packages.

1.5. UPGRADING TO THE RED HAT JBOSS CORE SERVICES APACHE HTTP SERVER 2.4.29

**NOTE**

Where a Red Hat JBoss Core Services Apache HTTP Server 2.4.23 or earlier was installed from RPMs packages using `yum`, the Apache HTTP Server can be upgraded with `yum upgrade`.

For systems where an earlier version of the Red Hat JBoss Core Services Apache HTTP Server was installed from a .zip archive, upgrading to the Apache HTTP Server 2.4.29 requires:

1. Installing the Apache HTTP Server 2.4.29.
2. Setting up the Apache HTTP Server 2.4.29.
3. Removing the earlier version of Apache HTTP Server.

**Prerequisites**

- Root user access (Red Hat Enterprise Linux and Solaris systems)
- Administrative access (Windows Server)
- A system where the Red Hat JBoss Core Services Apache HTTP Server 2.4.23 or earlier was installed from a .zip archive.

**Procedure**

For systems using the Red Hat JBoss Core Services Apache HTTP Server 2.4.23, the recommended procedure for upgrading to the Apache HTTP Server 2.4.29 is:

1. Shutdown any running instances of Red Hat JBoss Core Services Apache HTTP Server 2.4.23.
2. Backup the Red Hat JBoss Core Services Apache HTTP Server 2.4.23 installation and configuration files.
3. Install the Red Hat JBoss Core Services Apache HTTP Server 2.4.29 using the .zip installation method for the current system (see Additional Resources below).
4. Migrate your configuration from the Red Hat JBoss Core Services Apache HTTP Server version 2.4.23 to version 2.4.29.

**NOTE**

The Apache HTTP Server configuration files may have changed since the Apache HTTP Server 2.4.23 release. It is recommended that you update the 2.4.29 version configuration files, rather than overwrite them with the configuration files from a different version (such as Apache HTTP Server 2.4.23).

5. Remove the Red Hat JBoss Core Services Apache HTTP Server 2.4.23 root directory.

**Additional Resources**

- For installation instructions for Red Hat Enterprise Linux systems, see: Installing JBoss Core Services Apache HTTP Server on Red Hat Enterprise Linux: Installation using .zip archives.
- For installation instructions for Microsoft Windows systems, see: Installing JBoss Core Services Apache HTTP Server on Microsoft Windows.
- For installation instructions for Solaris systems, see: Installing Apache HTTP Server on Solaris.
CHAPTER 2. INSTALLING JBOSS CORE SERVICES APACHE HTTP SERVER ON RED HAT ENTERPRISE LINUX

2.1. ZIP INSTALLATION

2.1.1. Prerequisites

The following packages are required to run Red Hat JBoss Core Services Apache HTTP Server 2.4.29 on Red Hat Enterprise Linux:

- elinks
- krb5-workstation
- mailcap

To install these prerequisites on Red Hat Enterprise Linux, issue the following command as the root user:

```
# yum install elinks krb5-workstation mailcap
```

2.1.2. Download and Extract Apache HTTP Server

To install Apache HTTP Server, download and extract the installation ZIP files. Installation can be performed by non-root users if the user account has write access to the intended installation directory.

1. Open a browser and log in to the Red Hat Customer Portal JBoss Software Downloads page.
2. Select Apache HTTP Server in the Product drop-down menu.
3. Select the correct JBoss Core Services version from the Version drop-down menu.
4. Find Red Hat JBoss Core Services Apache HTTP Server in the list, ensuring that you select the correct platform and architecture for your system, and click the Download link.
5. Extract the downloaded ZIP file to your installation directory.

**NOTE**

We recommend that you install Apache HTTP Server in the /opt/ directory.

The jbcshttpd24-2.4/httpd directory created by extracting the ZIP archive is the top-level directory for Apache HTTP Server. This is referred to in this documentation as HTTPD_HOME.

2.1.3. Configuring the Apache HTTP Server Installation

Some configuration is required before running JBoss Core Services Apache HTTP Server. This section includes the following configuration procedures:

- Creating an Apache User
- Disabling/Enabling SSL Support
- Running the Apache HTTP Server Post-Installation Script
Creating an Apache User
Follow this procedure to create the apache user and its parent group:

As the root user:

1. On a command line, change directory to $HTTPD_HOME$.

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

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

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

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

4. From $HTTPD_HOME$, issue the following command to assign the ownership of the Apache directories to the apache user to allow the user to run Apache HTTP Server:

   ```bash
   # chown -R apache:apache *
   ```

   You can use `ls -l` to verify that the apache user is the owner of the directory.

Disabling/Enabling SSL Support
Apache HTTP Server supports SSL by default, but it can be disabled. Follow this procedure to disable or re-enable SSL support.

1. Go to the $HTTPD_HOME/conf.d/$ directory and rename the SSL configuration file:
   a. To disable SSL, rename `ssl.conf` to `ssl.conf.disabled`.
   b. To re-enable SSL, rename `ssl.conf.disabled` to `ssl.conf`.

Running the Apache HTTP Server Post-Installation Script

1. On a command line, change to the $HTTPD_HOME$ directory (root user access not required).

2. Issue the following command:

   ```bash
   ./postinstall
   ```

2.1.4. Starting Apache HTTP Server
To start Apache HTTP Server, on a command line as root user, change to $HTTPD_HOME/sbin/ and issue the following command:

   ```bash
   ./apachectl start
   ```

2.1.5. Stopping Apache HTTP Server
To stop Apache HTTP Server, on a command line as root user, change to $HTTPD_HOME/sbin/ and issue the following command:

   ```bash
   ./apachectl stop
   ```
2.1.6. Running Apache HTTP Server without root access on Red Hat Enterprise Linux (ZIP installation)

To run Apache HTTP Server as a non-root user, such as the apache user:

1. Stop all instances of Apache HTTP Server:
   ```
   pkill httpd
   ```

2. Set the http listen port to higher than 1024 in `HTTPD_HOME/conf/httpd.conf`:
   ```
   Listen 2080
   ServerName <hostname>:2080
   ```

3. Set the https listen port to higher than 1024 in `HTTPD_HOME/conf.d/ssl.conf`:
   ```
   Listen 2443
   ```

4. Change the ownership of the logs directory:
   ```
   chown -R apache:apache HTTPD_HOME/logs/
   ```

5. Change the ownership of the run directory:
   ```
   chown -R apache:apache HTTPD_HOME/var/run/
   ```

6. Verify that `httpd` is only running under the apache user, not the root and apache users:
   ```
   $ ps -eo euser,egroup,comm | grep httpd
   apache   apache   httpd
   apache   apache   httpd
   apache   apache   httpd
   ...
   ```

   **IMPORTANT**

   To prevent unauthorized access or modification of files and directories by website users and to prevent unwanted changes to Apache HTTP Server configuration files, limit the file permissions of the apache user and enable SELinux.

---

2.2. RPM INSTALLATION

Installing JBoss Core Services Apache HTTP Server from RPM packages installs Apache HTTP Server as a service. The RPM installation option is available for Red Hat Enterprise Linux 6, Red Hat Enterprise Linux 7 and Red Hat Enterprise Linux 8

2.2.1. Using mod_jk, mod_cluster, mod_rt, and mod_bmx with RHEL 8

This section cover proper procedure to install and configure the JBCS modules; mod_jk, mod_cluster, mod_rt, and mod_bmx on the RHEL 8 OS.
Installing httpd
To install httpd run the following command with root access:

```bash
$ yum install httpd
```

Installing Modules
To install mod_jk, mod_cluster, mod_rt, and mod_bmx, run the following commands with root access:

```bash
$ yum install jbcs-httpd24-mod_jk-ap24
$ yum install jbcs-httpd24-mod_cluster-native
$ yum install jbcs-httpd24-mod_bmx
$ yum install jbcs-httpd24-mod_rt
```

For RHEL-8 httpd has its BaseOS modules directory in `/usr/lib64/httpd/modules`. For the time being, JBCS modules are located in `/opt/rh/jbcs/root/usr/lib64/httpd/modules` and follow all JBCS rules in regards to naming, directories, and prefixes. This includes mod_jk, mod_cluster, mod_rt, and mod_bmx.

If you want to use these modules, create or modify configuration file to add `LoadModule` command, for example:

```bash
LoadModule jk_module /opt/rh/jbcs/root/usr/lib64/httpd/modules/mod_jk.so
```

**NOTE**
- `mod_proxy_balancer` **MUST** be disabled when `mod_proxy_cluster` is used.
- `mod_proxy` **MUST** be enabled when `mod_proxy_cluster` is used
- If one needs `mod_proxy_cluster` to use AJP, `proxy_ajp_module` must be enabled

Alternatively you may include the directory of the installed JBCS modules in the `JBCS_HOME/httpd/conf.d` directory.

### 2.2.2. Installing the Apache HTTP Server from RPM Packages

Before downloading and installing the RPM packages, you must register your system with Red Hat Subscription Management and subscribe to the respective Content Delivery Network (CDN) repositories.

For information on registering Red Hat Enterprise Linux, see:

- Configuring the Subscription Service for Red Hat Enterprise Linux 6
- The Subscription Manager for Red Hat Enterprise Linux 7
- The Subscription Manager for Red Hat Enterprise Linux 8
Attaching subscriptions to Red Hat Enterprise Linux (if required)

If the system does not have a subscription attached that provides the Apache HTTP Server:

1. Log in to the Red Hat Subscription Manager.
2. Click on the Systems tab.
3. Click on the Name of the system to add the subscription to.
4. Change from the Details tab to the Subscriptions tab, then click Attach Subscriptions.
5. Select the check box beside the subscription to attach, then click Attach Subscriptions.

NOTE

To verify that a subscription provides the required CDN repositories:

1. Log in to: https://access.redhat.com/management/subscriptions.
2. Click the Subscription Name.
3. Under Products Provided, you require:
   - Red Hat JBoss Core Services.

Installing the Apache HTTP Server from RPM packages using YUM

1. On a command line, subscribe to the Apache HTTP Server CDN repositories for your operating system version using subscription-manager:

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

   - For Red Hat Enterprise Linux 6:
     - jb-coreservices-1-for-rhel-6-server-rpms
   - For Red Hat Enterprise Linux 7:
     - jb-coreservices-1-for-rhel-7-server-rpms

2. Run the following command as the root user to install the Apache HTTP Server:

   ```shell
   # yum groupinstall jbcs-httpd24
   ```

   NOTE

   For complete instructions on installing and configuring HTTPD on RHEL 8, please see this link

2.2.3. Configuring the Apache HTTP Server Installation (RPM Installation)

Before starting an RPM installation of JBoss Core Services Apache HTTP Server, there are some optional configurations you can perform. This section includes the following configuration procedures:

- Removing SSL Support
Removing SSL Support
Apache HTTP Server supports SSL by default, but it can be removed. To remove SSL support, remove the mod_ssl package.

1. At a shell prompt, run the following command as the root user:

```bash
# yum remove jbcs-httpd24-mod_ssl
```

**NOTE**
For complete instructions on installing and configuring HTTPD on RHEL 8, please see this link

2.2.4. Starting Apache HTTP Server
In a shell prompt as the root user, start the Apache HTTP Server service:

- For Red Hat Enterprise Linux 6:

  ```bash
  # service jbcs-httpd24-httpd start
  ```

- For Red Hat Enterprise Linux 7:

  ```bash
  # systemctl start jbcs-httpd24-httpd.service
  ```

**NOTE**
For complete instructions on installing and configuring HTTPD on RHEL 8, please see this link

2.2.5. Stopping Apache HTTP Server
In a shell prompt as the root user, stop the Apache HTTP Server service:

- For Red Hat Enterprise Linux 6:

  ```bash
  # service jbcs-httpd24-httpd stop
  ```

- For Red Hat Enterprise Linux 7:

  ```bash
  # systemctl stop jbcs-httpd24-httpd.service
  ```

**NOTE**
For complete instructions on installing and configuring HTTPD on RHEL 8, please see this link

2.2.6. Configuring Apache HTTP Server to Start at Boot
Use the following command to enable the Apache HTTP Server service to start at boot.

- For Red Hat Enterprise Linux 6:
2.3. SELinux Policies

2.3.1. SELinux Policy Information

The Security-Enhanced Linux (SELinux) security model is enforced by the kernel and ensures applications have limited access to resources such as file system locations and ports. This helps ensure that the errant processes (either compromised or poorly configured) are restricted and in some cases prevented from running.

The following table contains information about the SELinux policies provided in the jbcs-httpd2.4-httpd-selinux packages.

<table>
<thead>
<tr>
<th>Name</th>
<th>Port Information</th>
<th>Policy Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>mod_cluster</td>
<td>Two ports ($6666$ for TCP and $23364$ for UDP) are added for httpd_port_t to allow the httpd process to use them.</td>
<td>A post installation script configures the context mapping for /var/cache/mod_cluster to enable the httpd process to write at this location.</td>
</tr>
</tbody>
</table>

For more information about using SELinux and other Red Hat Enterprise Linux security information, see the Red Hat Enterprise Linux Security Guide.

2.3.2. SELinux Policies for an RPM Installation

SELinux policies for Apache HTTP Server are provided by the jbcs-httpd2.4-httpd-selinux package available in the jb-coreservices-1-for-rhel-7-server-rpms and jb-coreservices-1-for-rhel-6-server-rpms Content Delivery Network (CDN) repositories.

To enable SELinux policies on Apache HTTP Server, install the jbcs-httpd2.4-httpd-selinux package for the version of Red Hat Enterprise Linux in use.

2.3.3. SELinux Policies for a ZIP Installation

**IMPORTANT**

By default, the SELinux policy provided is not active and the Apache HTTP Server processes run in the unconfined_t domain. This domain does not confine the processes, and if you chose not to enable the SELinux policy provided, it is recommended that you restrict file access for the apache user to the files and directories required by the Apache HTTP Server runtime.

For this release, SELinux policies are provided in the ZIP packages. The .postinstall.selinux file is included in root Apache HTTP Server folder. If required, you can run the .postinstall.selinux script.
To install the SELinux policies for ZIP installations:

1. Install the `selinux-policy-devel` package:
   
   ```bash
   yum install -y selinux-policy-devel
   ```

2. Execute the `.postinstall.selinux` script:
   
   ```bash
   cd <httpd_home>
   sh .postinstall.selinux
   ```

3. Make and install the SELinux module:
   
   ```bash
   cd <httpd_home>/selinux/
   make -f /usr/share/selinux/devel/Makefile
   semodule -i jbcs-httpd24-httpd.pp
   ```

4. Apply the SELinux contexts for Apache HTTP Server:
   
   ```bash
   restorecon -r <httpd_home>
   ```

5. Add access permissions to the required ports for Apache HTTP Server:
   
   ```bash
   semanage port -a -t http_port_t -p tcp 6666
   semanage port -a -t http_port_t -p udp 23364
   ```

6. Start the Apache HTTP Server service:
   
   ```bash
   <httpd_home>/sbin/apachectl start
   ```

7. Check the context of the running process expecting `httpd_t`:
   
   ```bash
   $ ps -eZ | grep httpd | head -n1
   unconfined_u:unconfined_r: httpd_t:s0-s0:c0.c1023 2864 ? 00:00:00 httpd
   ```

8. To verify the contexts of the httpd directories, for example:
   
   ```bash
   ls -lZ <httpd_home>/logs/
   ```
3.1. DOWNLOAD AND EXTRACT APACHE HTTP SERVER

To install Apache HTTP Server, download and extract the installation ZIP files. Installation can be performed by non-root users if the user account has write access to the intended installation directory.

1. Open a browser and log in to the Red Hat Customer Portal JBoss Software Downloads page.
2. Select Apache HTTP Server in the Product drop-down menu.
3. Select the correct JBoss Core Services version from the Version drop-down menu.
4. Find Red Hat JBoss Core Services Apache HTTP Server in the list, ensuring that you select the correct platform and architecture for your system, and click the Download link.
5. Extract the downloaded ZIP file to your installation directory.

**NOTE**

We recommend that you install Apache HTTP Server in the C:\Program Files directory.

The jbcs-httpd24-2.4 directory created by extracting the ZIP archive is the top-level directory for Apache HTTP Server. This is referred to in this documentation as HTTPD_HOME.

3.2. CONFIGURING THE APACHE HTTP SERVER INSTALLATION

Some configuration is required before running JBoss Core Services Apache HTTP Server. This section includes the following configuration procedures:

- Running the Apache HTTP Server Post-Installation Script
- Installing the Apache HTTP Server Service
- Configuring Folder Permissions for the Apache HTTP Server Service
- Disabling/Enabling SSL Support

**Running the Apache HTTP Server Post-Installation Script**

1. At the Command Prompt as an administrative user, change to the HTTPD_HOME\etc directory.
2. Run the following command:
   ```
   call postinstall.httpd.bat
   ```

**Installing the Apache HTTP Server Service**
NOTE

By default, Apache HTTP Server is configured to use port 80. If you have Microsoft IIS installed, ensure that you disable or reconfigure it to avoid port conflicts:

- Stop the World Wide Web... service, and change the Startup Type to Manual.
- Configure IIS to use different ports.

Alternatively, you can edit httpd.conf before installing the Apache HTTP Server service and change Listen to a port that does not conflict with the IIS ports.

1. At the Command Prompt as an administrative user, change to the HTTPD_HOME\bin directory.

2. Install the Apache HTTP Server service with the following command:

   ```bash
   httpd -k install
   ```

   A Firewall security dialog prompt may appear asking for networking access for Apache HTTP Server. Click Allow to access this service from the network.

Configuring Folder Permissions for the Apache HTTP Server Service

Follow this procedure to ensure that the account used to run the service has full control over the HTTPD_HOME folder and all of its subfolders:

1. Right-click the HTTPD_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.


10. Select the check box next to the Replace all existing inheritable permissions on all descendants with inheritable permissions from this object option.

11. Click OK through all the open folder property windows to apply the settings.

Disabling/Enabling SSL Support

Apache HTTP Server supports SSL by default, but it can be disabled. Follow this procedure to disable or re-enable SSL support.

1. Go to the HTTPD_HOME\conf.d directory and rename the SSL configuration file:
   a. To disable SSL, rename ssl.conf to ssl.conf.disabled.
b. To re-enable SSL, rename `ssl.conf.disabled` to `ssl.conf`.

3.3. STARTING APACHE HTTP SERVER

You can start the Apache HTTP Server service from the Command Prompt, or with the Computer Management tool.

Starting Apache HTTP Server Using the Command Prompt

1. At the Command Prompt as an administrative user, start the Apache HTTP Server service with the following command:

   ```
   net start Apache2.4
   ```

Starting Apache HTTP Server Using the Computer Management Tool

1. Go to Start → Administrative Tools → Services.
2. In the Services list, right-click the `httpd` service and click Start.

3.4. STOPPING APACHE HTTP SERVER

You can stop the Apache HTTP Server service from the Command Prompt, or with the Computer Management tool.

Stopping Apache HTTP Server Using the Command Prompt

1. At the Command Prompt as an administrative user, stop the Apache HTTP Server service with the following command:

   ```
   net stop Apache2.4
   ```

Stopping Apache HTTP Server Using the Computer Management Tool

1. Go to Start → Administrative Tools → Services.
2. In the Services list, right-click the `httpd` service and click Stop.
CHAPTER 4. INSTALLING APACHE HTTP SERVER ON SOLARIS

4.1. DOWNLOAD AND EXTRACT APACHE HTTP SERVER

To install Apache HTTP Server, download and extract the installation ZIP files. Installation can be performed by non-root users if the user account has write access to the intended installation directory.

1. Open a browser and log in to the Red Hat Customer Portal JBoss Software Downloads page.
2. Select Apache HTTP Server in the Product drop-down menu.
3. Select the correct JBoss Core Services version from the Version drop-down menu.
4. Find Red Hat JBoss Core Services Apache HTTP Server in the list, ensuring that you select the correct platform and architecture for your system, and click the Download link.
5. Extract the downloaded ZIP file to your installation directory.

NOTE

We recommend that you install Apache HTTP Server in the /opt/ directory.

The jbcs-httpd24-2.4 directory created by extracting the ZIP archive is the top-level directory for Apache HTTP Server. This is referred to in this documentation as HTTPD_HOME.

4.2. CONFIGURING THE APACHE HTTP SERVER INSTALLATION

Some configuration is required before running JBoss Core Services Apache HTTP Server. This section includes the following configuration procedures:

- Running the Apache HTTP Server Post-Installation Script
- Disabling/Enabling SSL Support

Running the Apache HTTP Server Post-Installation Script

1. On a command line as the root user, change to the HTTPD_HOME/etc directory.
2. Run the following command:

   # sh .postinstall.httpd

   The post-installation script creates the Apache user and group.

Disabling/Enabling SSL Support

Apache HTTP Server supports SSL by default, but it can be disabled. Follow this procedure to disable or re-enable SSL support.

1. Go to the HTTPD_HOME/conf.d/ directory and rename the SSL configuration file:
   a. To disable SSL, rename ssl.conf to ssl.conf.disabled.
   b. To re-enable SSL, rename ssl.conf.disabled to ssl.conf.
4.3. STARTING APACHE HTTP SERVER

To start Apache HTTP Server, on a command line as root user, change to `HTTPD_HOME/sbin/` and issue the following command:

```
./apachectl start
```

4.4. STOPPING APACHE HTTP SERVER

To stop Apache HTTP Server, on a command line as root user, change to `HTTPD_HOME/sbin/`, and issue the following command:

```
./apachectl stop
```

4.5. RUNNING APACHE HTTP SERVER WITHOUT ROOT ACCESS ON SOLARIS (ZIP INSTALLATION)

To run Apache HTTP Server as a non-root user, such as the `apache` user:

1. Stop all instances of Apache HTTP Server:
   ```
   pkill httpd
   ```

2. Set the `http` listen port to higher than 1024 in `HTTPD_HOME/conf/httpd.conf`:
   ```
   Listen 2080
   ServerName <hostname>:2080
   ```

3. Set the `https` listen port to higher than 1024 in `HTTPD_HOME/conf.d/ssl.conf`:
   ```
   Listen 2443
   ```

4. Change the ownership of the `logs` directory:
   ```
   chown -R apache:apache HTTPD_HOME/logs/
   ```

5. Change the ownership of the `run` directory:
   ```
   chown -R apache:apache HTTPD_HOME/var/run/
   ```

6. Verify that `httpd` is only running under the `apache` user, not the `root` and `apache` users:
   ```
   $ ps -eo euser,egroup,comm | grep httpd
   apache apache httpd
   apache apache httpd
   apache apache httpd
   ...
IMPORTANT

To prevent unauthorized access or modification of files and directories by website users and to prevent unwanted changes to Apache HTTP Server configuration files, limit the file permissions of the **apache** user.
CHAPTER 5. ENABLING HTTP/2 FOR THE JBOSS CORE SERVICES HTTP SERVER

The Hypertext Transfer Protocols are standard methods of transmitting data between applications (such as servers and browsers) over the internet. HTTP/2 improves on HTTP/1.1 by providing enhancements such as:

- header compression - reducing the size of the header transmitted by omitting implied information, and
- multiple requests and responses over a single connection - using binary framing to break down response messages, as opposed to textual framing.

Using HTTP/2 with the Red Hat JBoss Core Services Apache HTTP Server:

- is supported for encrypted connections using Transport Layer Security (TLS) (SSLEnabled="true"), indicated by the h2 keyword when enabled.
- is not supported for unencrypted connections using the Transmission Control Protocol (TCP) indicated by the h2c keyword when enabled.

NOTE

HTTP/2 is not available for web servers using the Multi-Processing Module prefork modules/mod_mpm_prefork.so

Prerequisites

- Root user access (Red Hat Enterprise Linux and Solaris systems)
- Administrative access (Windows Server)
- Red Hat JBoss Core Services Apache HTTP Server 2.4.23 or higher
- Modules required:
  - ssl_module modules/mod_ssl.so
  - http2_module modules/mod_http2.so

Procedure

Enable HTTP/2 for an Apache HTTP Server:

1. Add the http2_module to HTTP_HOME/conf.modules.d/00-base.conf:

   ```
   ...
   LoadModule http2_module modules/mod_http2.so
   ```

2. Add the h2 protocol in HTTP_HOME/conf/httpd.conf.

   - To enable HTTP/2 support for a virtual host, add the h2 protocol to the virtual host configuration,

   - To enable HTTP/2 support for all server connections, add the h2 protocol to the ‘Main’ server configuration section of httpd.conf.
3. Update the Secure Socket Layer (SSL) configuration in `HTTP_HOME/conf.d/ssl.conf`.
   
a. Ensure the `SSLEngine` directive is set to enabled (the SSL Engine is enabled by default):
   ```
   SSLEngine on
   ```

b. Update the `SSLProtocol` directive to disable the `SSLv2` and `SSLv3` protocols, forcing connections to use the Transport Layer Security (TLS) Protocols:
   ```
   SSLProtocol all -SSLv2 -SSLv3
   ```

c. Update the `SSLCipherSuite` directive to specify which SSL ciphers can with the Apache HTTP Server.
   For example:
   ```
   ```

   **NOTE**
   For information on the SSL module and the supported directives, see: [Apache HTTP Server Documentation Version 2.4 - Modules: Apache Module mod_ssl](https://httpd.apache.org/docs/2.4/mod/mod_ssl.html)

4. Restart the Red Hat JBoss Core Services Apache HTTP Server as the root user, to apply the changed configuration.
   
a. For SysV (Red Hat Enterprise Linux 6) users:
   ```
   # service jbcs-httpd24-httpd restart
   ```

b. For systemd (Red Hat Enterprise Linux 7) users:
   ```
   # systemctl restart jbcs-httpd24-httpd.service
   ```

c. For Red Hat Enterprise Linux users running Red Hat JBoss Core Services using `apachectl`:
   ```
   # HTTP_HOME/sbin/apachectl restart
   ```

   **IfModule** http2_module
   ```
   Protocols h2 http/1.1
   ProtocolsHonorOrder on
   ```
   **IfModule**

   **CHAPTER 5. ENABLING HTTP/2 FOR THE JBOSS CORE SERVICES HTTP SERVER**

21
d. For Solaris users:

```bash
# HTTP_HOME/sbin/apachectl restart
```

e. For Windows Server users:

```bash
# net restart Apache2.4
```

### Next Steps

Verify that HTTP/2 is enabled by reviewing the Apache HTTP Server logs or by using the `curl` command:

- Access the server from a browser or using `curl`, then check the SSL/TLS access or request logs (`HTTP_HOME/logs/ssl_access_log` or `HTTP_HOME/logs/ssl_access_log`) to verify that the connection is configured to support HTTP/2:

```bash
$ grep 'HTTP/2' HTTP_HOME/logs/ssl_request_log
```

```
[26/Apr/2018:06:44:45 +0000] 172.17.0.1 TLSv1.2 AES128-SHA "HEAD /html-single/index.html HTTP/2" -
```

```bash
$ grep 'HTTP/2' HTTP_HOME/logs/ssl_access_log
```

```
172.17.0.1 - - [26/Apr/2018:06:44:45 +0000] "HEAD /html-single/index.html HTTP/2" 200 -
```

- Or verify using `curl` (for versions of `curl` that support HTTP2):

```
NOTE
```

The `curl` package provided with Red Hat Enterprise Linux 7 or earlier does not support HTTP/2. To check `curl` for HTTP/2 support:

```bash
$ curl -V
```

```
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
```

- For example, when the HTTP/2 protocol is inactive:

```bash
$ curl -I http://<JBCS_httpd_server>:80/<test.html>
```

```
HTTP/1.1 200 ...
```

- But if the HTTP/2 protocol is active, `curl` returns:

```bash
```

```
HTTP/2 200 ...
```
Where:

- `<JBCS_httpd_server>` is the URI of the server (such as example.com),
- the port number is dependent on your configuration,
- `<test.html>` is any html page for testing the configuration (not provided), and

**Additional Resources**

- For additional information on using HTTP/2, see: Apache HTTP Server Documentation Version 2.4 - How-To / Tutorials: HTTP/2 guide.
- For information on SSL configuration, see: Apache HTTP Server Documentation Version 2.4 - SSL/TLS Strong Encryption: How-To.
- For information on the HTTP/2 module and the supported directives, see: Apache HTTP Server Documentation Version 2.4 - Modules: Apache Module mod_http2.
- For information on the SSL module and the supported directives, see: Apache HTTP Server Documentation Version 2.4 - Modules: Apache Module mod_ssl.
CHAPTER 6. MONITORING APACHE HTTP SERVER WITH JBOSS OPERATIONS NETWORK (ON)

To allow JBoss ON to monitor JBoss Core Services Apache HTTP Server, you must enable the mod_bmx modules.

1. As the apache user, enable the mod_bmx modules by creating
   \texttt{HTTPD\_HOME/conf.d/mod\_bmx.conf} with the following contents:
   
   \begin{verbatim}
   LoadModule bmx_module modules/mod_bmx.so
   LoadModule bmx_status_module modules/mod_bmx_status.so
   LoadModule bmx_vhost_module modules/mod_bmx_vhost.so
   \end{verbatim}

2. In \texttt{HTTPD\_HOME/conf/httpd.conf}, add a handler for mod_bmx by appending your handler definition. For example:

   \begin{verbatim}
   <Location /bmx>
   SetHandler bmx-handler
   </Location>
   \end{verbatim}

   You may want to restrict access to this URL to a specific virtual host or use password authentication.

3. Restart the Apache HTTP Server. You can access \texttt{http://SERVER\_ADDRESS:PORT/bmx} in a web browser to verify that mod_bmx is working correctly.

4. Apache HTTP Server will now be discoverable for JBoss ON. In JBoss ON, you can import it from the Discovery Queue.
   Consult the JBoss ON documentation for further details on monitoring and managing Apache HTTP Server.

Revised on 2019-05-07 17:08:15 UTC