Red Hat JBoss Web Server 3.1

3.1.0 Release Notes

For Use with Red Hat JBoss Web Server 3.1

Last Updated: 2018-08-30

Red Hat JBoss Web Server 3.1 3.1.0 Release Notes

For Use with Red Hat JBoss Web Server 3.1

Legal Notice

Abstract

These release notes contain important information related to Red Hat JBoss Web Server 3.1.

Table of Contents

CHAPTER 1. INTRODUCTION TO RED HAT JBOSS WEB SERVER 3.1

Welcome to Red Hat JBoss Web Server, formerly known as JBoss Enterprise Web Server. These Release Notes detail information about new features, enhancements, technical preview, known issues, and resolved issues. Use this document in conjunction with the entire JBoss Web Server 3.1 documentation suite, available on the Red Hat Customer Portal: https://access.redhat.com/documentation/en/red-hat-jboss-web-server/.

1.1. ABOUT RED HAT JBOSS WEB SERVER

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

CHAPTER 2. NEW FEATURES AND ENHANCEMENTS

2.1. APACHE HTTP SERVER SEPARATED FROM TOMCAT

The Apache HTTP Server distribution is now shared between the JWS and JBoss Core Services entitlements. The shared distribution can be downloaded in ZIP from Apache HTTP Server download page on the support portal. The RPM distribution of HTTP must be consumed from the JBCS channel, while the Tomcat servers will continue to be delivered in the JWS3 channel.

Installation instructions for the Apache HTTP Server are provided in the Apache HTTP Server Installation Guide. You should refer to that guide for instructions for ZIP and RPM setup on the set of supported operating systems.

Maintenance for the Apache HTTP Server and the Tomcat servers will no longer be coordinated in JWS minor and micro releases. The HTTP server and the Tomcat servers will receive independent updates intended to provide more timely fixes for security and other high priority defect fixes.

2.2. TRANSITION FROM HTTPD24 (JWS3 CHANNEL) TO JBCS-HTTPD24-HTTPD (JBCS CHANNEL)

To install **`httpd`** with JWS 3.1.0, you need to subscribe and enable the JBCS channel. The **`httpd`** package has moved from the JWS channel to the JBCS channel. If you are using **`httpd`**, then migrate from the **`httpd24`** package in JWS to the JBCS software collections new **`jbcs-httpd24-httpd`** package.

2.3. TOMCAT-NATIVE DEPENDENCIES AVAILABLE IN THE JBCS CHANNEL

The **`tomcat-native`** package requires the **`jbcs-httpd24-httpd-libs`** and **`jbcs-httpd24-openssl`** packages, which are available only in the JBCS channel. To access them, you have to subscribe and enable the JBCS channel.

2.4. TOMCAT

- Inclusion of the latest available version of Tomcat 8.0.36.
- Inclusion of the latest available version of Tomcat 7.0.70.
- Replaced the existing **`init`** scripts for Tomcat 7 and Tomcat 8 with **`systemd`** units on Red Hat Enterprise Linux 7.

2.5. USING A PASSWORD VAULT WITH RED HAT JBOSS WEB SERVER 3.1

A password vault is used to mask passwords and other sensitive strings, and store them in an encrypted Java keystore. This allows you to eliminate storing clear-text passwords in your Tomcat configuration files, as Tomcat can lookup passwords and other sensitive strings from a keystore using the vault.

NOTE

For more information about using password vault, see Using a Password Vault with Red Hat JBoss Web Server 3.1.

2.6. SELINUX POLICIES IN RHEL ZIP FOR TOMCAT

In this release, SELinux policies are provided in the ZIP packages. The 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 **.postinstall.selinux** file is included in each **tomcat** folder. If required, you can run the **postinstall.selinux** script.

To install the SELinux policies using ZIP:

1. Install the prerequisite packages:

   - **selinux-policy-devel**
   - Tomcat 7 or 8

2. Download and unzip the JWS Tomcat distribution from the JWS channel.

3. Execute the following commands:

```
cd $JWS_HOME/tomcat7  OR cd $JWS_HOME/tomcat8
sh .postinstall.selinux
cd selinux
make -f /usr/share/selinux/devel/Makefile
semodule -i tomcat7.pp OR semodule -i tomcat8.pp
cd $JWS_HOME
```

4. Start the Tomcat service.

```
bin/startup.sh
```

5. Check the context of the running process expecting **tomcat7_t**.

```
ps -eZf | grep tomcat | head -n1
```

6. To verify the contexts of the Tomcat log directory and so on.

```
ls -lZ tomcat7/logs/
```

2.7. SELINUX POLICIES IN RHEL RPM FOR TOMCAT

SELinux policies for each Tomcat are provided via their own Tomcat sub-packages: **tomcat7-selinux** and **tomcat8-selinux**. These packages are available in the JWS channel.

- To enable SELinux policies on Tomcat 7, install the **tomcat7-selinux** package.
- To enable SELinux policies on Tomcat 8, install the **tomcat8-selinux** package.

2.8. HIBERNATE

- Upgraded to Hibernate version 4.2.23.

2.9. MICROSOFT AZURE TESTING AND CERTIFICATION

- JBoss Web Server 3.1 has been tested and certified for Microsoft Azure.

2.10. UPDATED CGISERVLET TO RESOLVE HTTPOXY ISSUE

In this release, a CGIServlet fix is provided for the httpoxy issue, see CVE-2016-5388. The **`envHttpHeaders`** parameter is included in the CGIServlet to solve the httpoxy issue.

You can also configure the filter and valve to resolve the httpoxy issue. For more information about using the filter and valve, see HTTPoxy - Is my JBoss/tomcat affected?.

CHAPTER 3. TECHNOLOGY PREVIEW

WARNING

The following configurations and features are provided as technology previews only. They are not supported in a production environment, and may be subject to significant changes. See this note on the Red Hat Customer Portal on the support scope for Technology Preview features.

3.1. CONFIGURING THE WEB.XML TO USE THE SPNEGO AUTHENTICATION METHOD

The SPNEGO authentication method is available as a technical preview.

CHAPTER 4. VERIFIED CVES

- **JWS-431 CVE-2016-3092 Tomcat: Usage of vulnerable FileUpload package can result in denial of service**
  A denial of service vulnerability was identified in Commons FileUpload that occurred when the length of the multipart boundary was just below the size of the buffer (4096 bytes) used to read the uploaded file if the boundary was the typical tens of bytes long.

- **JWS-498 CVE-2016-6794 Provide a mechanism that enables the container to check if a component has been granted a given permission**
  It was discovered that when a SecurityManager is configured, Tomcat's system property replacement feature for configuration files could be used by a malicious web application to bypass the SecurityManager and read system properties that should not be visible.

- **JWS-500 CVE-2016-6797 Tomcat: unrestricted access to global resources**
  It was discovered that it was possible for a web application to access any global JNDI resource whether an explicit ResourceLink had been configured or not.

- **JWS-567 CVE-2016-0762 Tomcat: timing attack in Realm implementation**
  The Realm implementations did not process the supplied password if the supplied user name did not exist. This made a timing attack possible to determine valid user names. Note that the default configuration includes the LockOutRealm, which makes exploitation of this vulnerability harder.

- **JWS-568 CVE-2016-6796 Tomcat: security manager bypass via JSP Servlet config parameters**
  It was discovered that a malicious web application could bypass a configured SecurityManager via manipulation of the configuration parameters for the JSP Servlet.

- **JWS-569 CVE-2016-5018 Tomcat: security manager bypass via IntrospectHelper utility function**
  It was discovered that a malicious web application could bypass a configured SecurityManager via a Tomcat utility method that was accessible to web applications.

- **JWS-577 CVE-2016-6816 Tomcat: HTTP Request smuggling vulnerability due to permitting invalid character in HTTP requests**
  It was discovered that the code that parsed the HTTP request line permitted invalid characters. This could be exploited, in conjunction with a proxy that also permitted the invalid characters but with a different interpretation, to inject data into the HTTP response. By manipulating the HTTP response the attacker could poison a web-cache, perform an XSS attack and/or obtain sensitive information from requests other than their own.

- **JWS-578 CVE-2016-8735 Tomcat: Remote code execution vulnerability in JmxRemoteLifecycleListener**
  The JmxRemoteLifecycleListener was not updated to take account of Oracle's fix for CVE-2016-3427. JMXRemoteLifecycleListener is only included in EWS 2.x and JWS 3.x source distributions. If you deploy a Tomcat instance built from source, using the EWS 2.x, or JWS 3.x distributions, an attacker could use this flaw to launch a remote code execution attack on your deployed instance.

- **JWS-619 CVE-2016-8745 Tomcat: information disclosure due to incorrect Processor sharing**
  A bug was discovered in the error handling of the send file code for the NIO HTTP connector. This lead to the current Processor object being added to the Processor cache multiple times allowing information leakage between requests including, and not limited to, session ID and the response body.

- **JWS-538 CVE-2016-1240 Tomcat: unsafe chown of catalina.log in tomcat init script allows privilege escalation**
  It was reported that the Tomcat init script performed unsafe file handling, which could result in local privilege escalation.

- **JWS-490 CVE-2016-6325 Tomcat: tomcat writable config files allow privilege escalation**
  It was discovered that the Tomcat packages installed certain configuration files read by the Tomcat initialization script as writeable to the tomcat group. A member of the group or a malicious web application deployed on Tomcat could use this flaw to escalate their privileges.

- **JWS-701 CVE-2017-5647 Tomcat: Incorrect handling of pipelined requests when send file was used**
  A bug in the handling of the pipelined requests in Apache Tomcat 9.0.0.M1 to 9.0.0.M18, 8.5.0 to 8.5.12, 8.0.0.RC1 to 8.0.42, 7.0.0 to 7.0.76, and 6.0.0 to 6.0.52, when send file was used, results in the pipelined request being lost when send file processing of the previous request completed. This could result in responses appearing to be sent for the wrong request. For example, a user agent that sent requests A, B and C could see the correct response for request A, the response for request C for request B and no response for request C.

@TIM, MICHAL: THE MENTIONED JIRA IS NOT VISIBLE DUE TO THE CURRENT PERMISSION FOR THE JIRA. THIS NEEDS TO BE FIXED BY THE PM TEAM

- **JWS-668 CVE-2017-5648 Tomcat: Calls to application listeners did not use the appropriate facade object**
  While investigating bug 60718, it was noticed that some calls to application listeners in Apache Tomcat 9.0.0.M1 to 9.0.0.M17, 8.5.0 to 8.5.11, 8.0.0.RC1 to 8.0.41, and 7.0.0 to 7.0.75 did not use the appropriate facade object. When running an untrusted application under a SecurityManager, it was therefore possible for that untrusted application to retain a reference to the request or response object and thereby access and/or modify information associated with another web application.

@TIM, MICHAL: THE MENTIONED JIRA IS NOT VISIBLE DUE TO THE CURRENT PERMISSION FOR THE JIRA. THIS NEEDS TO BE FIXED BY THE PM TEAM

CHAPTER 5. VERIFIED AND RESOLVED ISSUES

5.1. VERIFIED ISSUES

The following JIRAs have been verified in this release:

- **JWS-162 - ASF Bug 57546 – Memory Leak in SecureNioChannel Tomcat8**
- **JWS-67 - ASF BZ 59859 – Windows Tomcat8 WebDAV move operation fails**
- **JWS-411 - RPM: dependency on base-os 'apr-util-ldap' package from '-optional' channel**
- **JWS-586 - Tomcat-vault INSTALL file is wrong**
- **JWS-572 - Possible async deadlocks**
- **JWS-516 - RHEL6 RPM: /usr/sbin/tomcat7 and /usr/sbin/tomcat8 don’t work**
- **JWS-499 - Compatibility with rewrite from httpd for non existing headers**
- **JWS-518 - CVE-2016-5388 Tomcat: CGI sets environmental variable based on user supplied Proxy request header**

5.2. RESOLVED ISSUES

The following issues have been included in this release, but have yet to be functionally tested:

- **JWS-531 - Tomcat7 manager: JMX Query returns quoted results**
- **JWS-501 - Add a limit (default 200) for the number of cookies allowed per request**

CHAPTER 6. KNOWN ISSUES FOR THE 3.1.0 RELEASE

JWS-365 - Socked bind failed on link-local (IPv6)

If you bind to a link-local IPv6 address, a SEVERE exception will be logged and the connector will fail to initialize.

Workaround

Bind the address in the configuration file without using the brackets. For instance, the following configuration snippet would bind correctly:

```
Listen ffff::ffff:ffff:ffff:ffff%3:80
```

JWS-132 - JON Tomcat: NullPointerException when Tomcat Web Application (WAR) config change

When monitoring JBoss Web Server using JBoss ON, a **`NullPointerException`** similar to below may be thrown when a Tomcat web application's configuration changed.

```
java.lang.NullPointerException
 at
org.rhq.plugins.jmx.MBeanResourceComponent.updateResourceConfiguration(M
BeanResourceComponent.java:532)
 at
org.jboss.on.plugins.tomcat.TomcatWarComponent.updateResourceConfigurati
on(TomcatWarComponent.java:950)
 at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
 at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.jav
a:57)
 at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessor
Impl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:606)
 at
org.rhq.core.pc.inventory.ResourceContainer$ComponentInvocation.call(Res
ourceContainer.java:759)
 at java.util.concurrent.FutureTask.run(FutureTask.java:262)
 at
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.jav
a:1145)
 at
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.ja
va:615)
 at java.lang.Thread.run(Thread.java:745)
```

The cause of this issue is currently under investigation.

Workaround

None

JWS-86- Upgrading Tomcat plugin, metric names not updated

When upgrading a RHQ 4.5 installation to 4.6, the Tomcat plugin may not be properly updated. In this situation the following exception is thrown:

```
java.lang.IllegalArgumentException: The property [name] marked as
required in the configuration definition of [Tomcat Connector] has no
attribute 'default'
```

The cause of this issue is currently under investigation.

Workaround

None

JWS-63 - Tomcat mod_cluster integration does not allow one to choose a connector

Configuring **`mod_cluster`** to use **`ws://`** with **`EnableWsTunnel`** (**`mod_proxy_wstunnel`**) requires Tomcat to use a **`http`** connector, and does not allow you to use an **`ajp`** connector. At the moment, Tomcat ignores this option. This issue is expected to be fixed in a future update.

Workaround

None

JWS-185 - (ASF BZ 53971) IPv6: address slowing down Tomcat on MS Windows

When an IPv6 address is used to start Tomcat 7 on Microsoft Windows (for example, **`<Connector port="8080" protocol="HTTP/1.1" address="::" />`**), this results in a significantly slowed down shutdown process when Tomcat is shutdown. During the shutdown process, the **`catalina.log`** file contains a warning similar to the following:

```
WARNING: Acceptor thread [http-0%3A0%3A0%3A0%3A0%3A0%3A0%3A0-8080-
Acceptor-0] failed to unlock. Forcing hard socket shutdown
```

The cause of this issue is currently under investigation.

Workaround

None

JWS-6 - hibernate c3p0 mchange-commons-java raises error on IBMJDK

If using **`mchange-commons-java`** in Hibernate c3p0 on an IBM JDK, a **`java.lang.ClassNotFoundException: org.slf4j.ILoggerFactory`** error may be thrown, such as the following:

```
java.lang.ClassNotFoundException: org.slf4j.ILoggerFactory
 at org.apache.catalina.loader.WebappClassLoader.loadClass(Unknown
Source)
 at org.apache.catalina.loader.WebappClassLoader.loadClass(Unknown
Source)
 at java.lang.Class.forNameImpl(Native Method)
 at java.lang.Class.forName(Class.java:199)
 at com.mchange.v2.log.MLog.findByClassnames(MLog.java:143)
 at com.mchange.v2.log.MLog.refreshConfig(MLog.java:73)
 at com.mchange.v2.log.MLog.<clinit>(MLog.java:51)
```

The cause of this issue is currently under investigation.

Workaround

You can work around this issue by setting the following property in your application:

```
System.setProperty("com.mchange.v2.log.MLog",
"com.mchange.v2.log.jdk14logging.Jdk14MLog");
```

JWS-645 - Document tomcat-native dependency on JBCS channel for jbcs-httpd24-httpd-libs

Installing the **`tomcat-native`** package fails due to the **`jbcs-httpd24-httpd-libs`** missing dependency.

Workaround

Ensure that the JBCS channel is enabled on the system when installing the **`tomcat-native`** package.

JWS-653 - Solaris, Windows - same tomcat-juli-adapters.jar and tomcat-juli.jar for Tomcat 7 and 8

The **`tomcat-juli`** and **`tomcat-juli-adapters`** JARs provided by the **`extras`** libraries in the **`jws-application-servers`** distribution for Windows and Solaris are incorrect for Tomcat 7. They are duplicates of the Tomcat 8 jars.

Workaround

None

JWS-521 - Add JSVC script to tomcat distribution

A JSVC wrapper was added to Tomcat on RHEL 6, however, the **`init`** script is missing. Additionally, the RHEL 7 **`systemd`** unit uses the system's JSVC binary rather than the JBCS provided binary.

Workaround

There is no workaround for the missing RHEL 6 init script. However, to workaround the RHEL 7 **`systemd`** unit issue, you have to manually update the path of the JSVC binary in the **`/usr/libexec/tomcat*/functions`** script to point to the correct **`/opt/rh/jbcs-httpd24/root/usr/bin/jsvc`** binary.

JWS-470 - RHEL 6 RPM Confusing service status output for tomcat7 and tomcat8 installed both on RHEL system

When running multiple instances of tomcat the **`init`** script status command may return incorrect information.

Workaround

None

JWS-401 - Tomcat does not properly parse spaces in JVM parameters/settings

Tomcat fails to start when using **`JAVA_OPTS`**, which have spaces in them.

Workaround

None

JWS-657 - tomcat-native installs RHEL apr in addition to jbcs-httpd24-httpd-libs

The installation of the **`tomcat-native`** RPM still incorrectly requires the installation of the **`apr`**/**`apr-util`** libraries distributed with RHEL. However, these libraries are not required as the **`apr`**/**`apr-utils`** versions distributed by JBCS are used. This requirement for the unused RHEL **`apr`**/**`apr-util`** libraries will be removed in a future update.

Workaround

If the RHEL `apr`/`apr-util` libraries are not installed, the system must be subscribed to the RHEL channel and the channel be enabled.

JWS-649 - Tomcat security-manager starting with errors in log

The manager and host-manager web applications fail to deploy when using the Security Manager.
Workaround

If you install the admin-webapps using the Security Manager, then move the `context.xml` from the manager and host-manager web applications, otherwise the admin applications will not deploy. Move from `CATALINA_HOME/webapps/[host-]manager/META-INF/context.xml` to `CATALINA_HOME/conf/Catalina/localhost/[host-]manager.xml`.

JWS-280 - Running async example servlet multiple times causes Tomcat crash

If the async example servlet is run multiple times, it causes Tomcat to crash.
Workaround

None

JWS-647 - Hibernate missing some JAR files in RPMs

If you install Hibernate using the RPM package from JWS3 channel, some files are missing.
Workaround

These missing Hibernate files are available in the ZIP distribution.

JWS-663 - service.bat for Tomcat 8 does not correctly set library path, natives not loaded

The `service.bat` for Tomcat 8 does not correctly set the library path and so the natives are not loaded.
Workaround

You have to manually append `C:\Program Files\jws-3.1\bin` to the system's PATH and start the Tomcat 8 service again.

JWS-662 - Windows, Cannot unzip Hibernate zip in GUI: Error: Path too long

On Windows, the Hibernate ZIP files displays an error if you unzip it using the user interface.
Workaround

Use PowerShell to unzip the Hibernate file.

```
$src = "C:\Users\ben\Documents\3.1.0-CR7\hibernate-dist-3.1.0-CR7.zip"
$dst = "C:\Users\ben\Documents\3.1.0-CR7\"
Add-Type -assembly "system.io.compression.filesystem"
[io.compression.zipfile]::ExtractToDirectory($src, $dst)
```

JWS-654 - RHEL 7 RPM Tomcat does not properly parse spaces in JVM parameters/settings

Tomcat fails to start when using `JAVA_OPTS`, which have spaces in them.
Workaround

Do not use spaces in parameters.

JWS-633 - ASF BZ 60683 - Tomcat throws NPE after startup with sec. manager on IBM JDK17

When using the Security Manager and IBM JDK, Tomcat may respond with a 500 status that may be triggered by the compilation of large JSP files. This is caused by an unhanded **`NullPointerException`**.

Workaround

You can precompile JSPs before deploying them to Tomcat.

JWS-666 - RHEL 7 RPM - TOMCAT_USER variable ignored

The **`TOMCAT_USER`** and **`TOMCAT_GROUP`** environment variables defined in the **`sysconfig`** and **`tomcat`** configuration files are ignored by the RHEL 7 systemd service units. This is a known issue for JWS 3.1.0 and is a limitation of the systemd design.

Workaround

In order to update the **`user/group`** owner of the Tomcat processes with the new systemd service unit on RHEL 7, you will need to update the **`User`** and **`Group`** properties of the service unit (**`/usr/lib/systemd/system/tomcat7.service`** and **`/usr/lib/systemd/system/tomcat8.service`**) rather than the **`sysconfig`** or tomcat **`conf`** files.