Red Hat JBoss Enterprise Application Platform 7.4-Beta

7.4.0 Beta Release Notes

For Use with Red Hat JBoss Enterprise Application Platform 7.4-Beta

Last Updated: 2021-03-23

Red Hat JBoss Enterprise Application Platform 7.4-Beta 7.4.0 Beta Release Notes

For Use with Red Hat JBoss Enterprise Application Platform 7.4-Beta

Legal Notice

Abstract

These release notes contain important information related to Red Hat JBoss Enterprise Application Platform 7.4-Beta.

Table of Contents

CHAPTER 1. NEW FEATURES AND ENHANCEMENTS

1.1. SECURITY

Support for automatic update of credentials in a credential store

Elytron now automates adding and updating a credential to a previously defined credential store when you configure a credential reference that specifies both the **store** and **clear-text** attributes.

With this update, you do not need to add a credential to an existing credential store before you can reference it from a **credential-reference**. The automated process reduces the number of steps you need to perform for referencing new credentials in different subsystems.

New role mapper regex-role-mapper in Elytron

Elytron now provides a new role mapper, **regex-role-mapper**, to define a regular expression (regex) based mapping of security roles.

You can use **regex-role-mapper** to translate a list of roles to simpler roles. For example:

- ***-admin** to **admin**
- ***-user** to **user**

With **regex-role-mapper**, you do not need to implement your own custom component to translate security roles.

Accessing IP address of remote client

You can now add the **source-address-role-decoder** role decoder to the **elytron** subsystem. By configuring this role decoder, you can gain additional information from a remote client when making authorization decisions.

The **source-address-role-decoder** extracts the IP address of a remote client and checks that it matches the IP address specified in the **pattern** attribute or the **source-address** attribute. If the IP address of the remote client matches the IP address specified in either attribute, the **roles** attribute then assigns roles to the user. When you have configured **source-address-role-decoder**, you can reference it in the **role-decoder** attribute of the **security domain**.

The aggregate-role-decoder role decoder

The **aggregate-role-decoder** consists of two or more role decoders. After each specified role decoder completes its operation, it adds roles to the **aggregate-role-decoder**.

You can use **aggregate-role-decoder** to make authorization decisions by adding role decoders that assign roles for a user. Further, **aggregate-role-decoder** provides you with a convenient way to aggregate the roles returned from each role decoder.

Using TLS protocol version 1.3 with JDK 11

Elytron now provides the ability to use Transport Layer Security (TLS) Protocol version 1.3 for JBoss EAP running against JDK 11.

TLS 1.3 is disabled by default. You can enable TLS 1.3 by configuring the new **cipher-suite-names** attribute in the SSL Context resource definition in the **elytron** subsystem.

Compared with TLS 1.2, you might experience reduced performance when running TLS 1.3 with JDK 11. Diminished performance might occur when a very large number of TLS 1.3 requests are being made. A system upgrade to a newer JDK version can improve performance. Test your setup with TLS 1.3 for performance degradation before enabling it in production.

Enable support for the TLS 1.3 protocol with the OpenSSL provider for TLS

JBoss EAP 7.4 includes support for the Transport Layer Security (TLS) protocol version 1.3. The use of TLS 1.3 protocol with the OpenSSL provider for TLS is disabled by default.

You can enable support for the TLS 1.3 protocol with the OpenSSL provider for TLS by configuring the **cipher-suite-names** attribute in the **ssl-context** configuration.

Compared with TLS 1.2, you might experience reduced performance when running TLS 1.3 with JDK 11. Diminished performance might occur when a very large number of TLS 1.3 requests are being made. A system upgrade to a newer JDK version can improve performance. Test your setup with TLS 1.3 for performance degradation before enabling it in production.

Using SSH credentials to connect to a remote Git SSH repository

With JBoss EAP 7.4, you can use SSH credentials to connect to a remote Git SSH repository. This repository can manage your server configuration data, properties files, and deployments.

You must use the **elytron** configuration file to specify SSH credentials. You can then start your standalone server instance and have a remote Git SSH repository manage your server configuration file history.

If necessary, you can generate SSH keys by using one of the following methods:

- The **elytron-tool.sh** script
- The OpenSSH command line

For information about connecting to a remote Git SSH repository, see Using a remote Git SSH repository.

New principal transformer added to the elytron subsystem

JBoss EAP 7.4 includes a new principal transformer, **case-principal-transformer**, in the **elytron** subsystem. You can use the **case-principal-transformer** to change a principal's username to either uppercase or lowercase characters.

Ability to automatically generate a self-signed certificate

With JBoss EAP 7.4, you can automatically generate a self-signed certificate.

Use a self-signed certificate only in a test environment. Do not use a self-signed certificate in a production environment.

To use this new feature, in the **undertow** subsystem, update the configuration of the **http-listener**.

```
batch
/subsystem=undertow/server=default-server/https-listener=https:undefine-attribute(name=security-realm)
/subsystem=undertow/server=default-server/https-listener=https:write-attribute(name=ssl-context,value=applicationSSC)
run-batch
reload
```

After you update the configuration, and if no keystore file exists, the first time JBoss EAP receives an HTTPS request, the system automatically generates a self-signed certificate. JBoss EAP logs a warning when a self-signed certificate is used.

Configuration of multiple security realms to support failover

With JBoss EAP 7.4, you can configure a failover security realm. If the security realm is not available, JBoss EAP uses the failover realm. The following code illustrates an example configuration:

```
<failover-realm name="myfailoverrealm" delegate-realm="LdapRealm" failover-realm="LocalRealm"
/>
```

Distributed identities across multiple security realms

With JBoss EAP 7.4, you can configure a distributed security realm, which sequentially invokes a list of configured realms until a realm with the identity is found. The following code illustrates an example configuration:

```
<distributed-realm name="mymainrealm" realms="realm1 realm2 realm3" />
```

RESTEasy client integration with the elytron subsystem

With JBoss EAP 7.4, RESTEasy clients are integrated with the **elytron** subsystem. With this integration, RESTEasy clients can use authentication information, such as credentials and SSL configurations, from an **elytron** client configuration file.

Access to external credentials over HTTP in the elytron subsystem

With JBoss EAP 7.4, JBoss EAP can authenticate a user based on credentials established externally when using HTTP authentication.

To use this capability, configure a security domain to use the External mechanism when authenticating users.

1.2. SERVER MANAGEMENT

Use a global directory to distribute shared libraries across deployments

In JBoss EAP 7.3 and earlier versions, you could not create and configure a global directory to distribute shared libraries across deployments running on a server. These capabilities have been added to the **ee** subsystem.

A global directory offers a better alternative to the global module approach. For example, if you want to change the name of a library listed in a global module, you must remove the global module, change the library's name, and then add the library to a new global module. If you change the name of a library that is listed in the global directory, you only need to restart the server to make the library name change available for all deployments.

Using a global directory is also a better solution if you want to share multiple libraries across deployed applications.

For more information, see Define global modules in the JBoss EAP *Configuration Guide*.

Support for read-only server configuration directories

In JBoss EAP 7.3 and earlier versions, servers fail to start if the configuration directory is configured as read-only. JBoss EAP 7.4 introduces the ability to use a read-only server configuration directory. If the configuration directory is read-only, include the **--read-only-server-config** switch in a command to start the server.

Ability to pass JBoss Module parameters

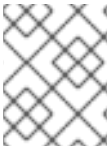
In the configuration files for JBoss EAP 7.3 and earlier versions, JBoss Modules did not include the ability to pass module parameters. In the script configuration files for JBoss EAP 7.4 you can now add a **MODULE_OPTS=-javaagent:my-agent.jar** environment variable to pass JBoss Module parameters.

You can use this capability when you previously were required to add the log manager on the boot class path.

1.3. MANAGEMENT CLI

Enhancement to the command CLI command

The CLI command **command** has a new **--node-child** argument that you can use to edit the properties or manage the operations of a specific child node.



NOTE

Before you use the **--node-child** argument, check that the child node exists in the management model.

Use the **command add --node-child --help** CLI command to view a description of the **--node-child** argument.

New role decoder added to the elytron subsystem

In JBoss EAP 7.4, you can use the management CLI to add the **source-address-role-decoder** role decoder to the **elytron** subsystem. By configuring this role decoder in the **mappers** element, you can gain additional information from a remote client when making authorization decisions.

You can configure the following attributes for **source-address-role-decoder**:

Attribute	Description
pattern	A regular expression that specifies the IP address of a remote client or the IP addresses of remote clients to match.
source-address	Specifies the IP address of the remote client.
roles	Provides the list of roles to assign to a user if the IP address of the remote client matches the values specified in the **pattern** attribute or the **source-address** attribute.

Exposing runtime statistics for managed executor services

In the previous JBoss EAP release, runtime statistics were not available for managed executor services in the **ee** subsystem.

You can now monitor the performance of managed executor services by viewing the runtime statistics generated with the new management CLI attributes. The following management CLI attributes have been added:

- **active-thread-count**: the approximate number of threads that are actively executing tasks
- **completed-task-count**: the approximate total number of tasks that have completed execution
- **hung-thread-count**: the number of executor threads that are hung
- **max-thread-count**: the largest number of executor threads

- **current-queue-size**: the current size of the executor's task queue
- **task-count**: the approximate total number of tasks that have been submitted for execution
- **thread-count**: the current number of executor threads

Using property replacement for permissions files

Users upgrading from JBoss EAP 6 to JBoss EAP 7 were unable to migrate file permissions in the Java policy file to the **permissions.xml** or **jboss-permissions.xml** files. It was not possible to use property replacement to migrate file permissions in the **permissions.xml** and **jboss-permissions.xml** files.

You can now use property replacement for the **permissions.xml** and **jboss-permissions.xml** files.

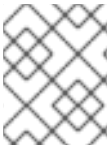
The property replacement for **jboss-permissions.xml** and **permissions.xml** files can be enabled or disabled using the **jboss-descriptor-property-replacement** and **spec-descriptor-property-replacement** attributes in the **ee** subsystem.

Configuring RESTEasy parameters

You can now use the JBoss EAP management CLI to change the settings for RESTEasy parameters. A global change applies the updated settings to new deployments as **web.xml** context parameters.

You can modify the settings of a parameter by using the **:write-attribute** operation with the **/subsystem=jaxrs** resource in the management CLI. For example:

```
/subsystem=jaxrs:write-attribute(name=resteasy-add-charset, value=false)
```



NOTE

When you change the settings of a parameter, the updated settings only apply to new deployments. Restart the server to apply the new settings to current deployments.

See the RESTEasy Configuration Parameters table for details about RESTEasy elements.

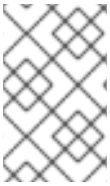
Configuring RESTEasy providers

In RESTEasy, certain built-in providers are enabled by default. You can now use the new RESTEasy parameter **resteasy.disable.providers** in the JBoss EAP management CLI to disable specific built-in providers.

The following example demonstrates how to disable the built-in provider **FileProvider**:

```
/subsystem=jaxrs:write-attribute(name=resteasy-disable-providers, value=
[org.jboss.resteasy.plugins.providers.FileProvider])
```

You can use the **resteasy.disable.providers** parameter with the pre-existing parameter **resteasy.use.builtin.providers** to customize a specific provider configuration that applies to all new deployments.



NOTE

When you change the settings of the **resteasy.disable.providers** parameter, the updated settings only apply to new deployments. Restart the server to apply the new settings to current deployments.

1.4. MANAGEMENT CONSOLE

New role decoder added to the elytron subsystem

In JBoss EAP 7.4, you can use the management console to add the **source-address-role-decoder** role decoder to the **elytron** subsystem. By configuring this role decoder in the **mappers** element, you gain additional information from a remote client when you make authorization decisions.

You can configure the following attributes for **source-address-role-decoder**:

Attribute	Description
pattern	A regular expression that specifies the IP address of a remote client or the IP addresses of remote clients to match.
source-address	Specifies the IP address of the remote client.
roles	Provides the list of roles to assign to a user if the IP address of the remote client matches the values specified in the **pattern** attribute or the **source-address** attribute.

1.5. LOGGING

The Apache Log4j2 API

In JBoss EAP 7.4, you can use an Apache Log4j2 API instead of an Apache Log4j API to send application logging messages to your JBoss LogManager implementation.

The JBoss EAP 7.4 release supports the Log4J2 API, but the release does not support the Apache Log4j2 Core implementation, **org.apache.logging.log4j:log4j-core**, or its configuration files.

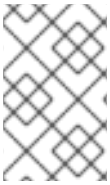
1.6. EJB3 SUBSYSTEM

Default global stateful session bean timeout value in the ejb3 subsystem

In the **ejb3** subsystem, you can now configure a default global timeout value for all stateful session beans (SFSBs) that are deployed on your server instance by using the **default-stateful-bean-session-timeout** attribute. This attribute is located in the JBoss EAP server configuration file. You can configure the attribute using the Management CLI.

Attribute behavior varies according to the server mode. For example:

- When running in the standalone server, the configured value gets applied to all SFSBs deployed on the application server.
- When running in the managed domain, all SFSBs that are deployed on server instances within server groups receive concurrent timeout values.



NOTE

When you change the global timeout value for the attribute, the updated settings only apply to new deployments. Reload the server to apply the new settings to current deployments.

By default, the attribute value is set at **-1** milliseconds, which means that deployed SFSBs are configured to never time out. However, you can configure two other types of valid values for the attribute, as follows:

- When the value is **0**, SFSBs are eligible for immediate removal by the **ejb** container.
- When the value is greater than **0**, the SFSBs remain idle for the specified time before they are eligible for removal by the **ejb** container.

You can still use the pre-existing **@StatefulTimeout** annotation or the stateful-timeout element, which is located in the **ejb-jar.xml** deployment descriptor, to configure the timeout value for an SFSB. However, setting such a configuration overrides the default global timeout value to the SFSB.

Forcing Jakarta Enterprise Beans timer refresh in database-data-store

You can now set the **wildfly.ejb.timer.refresh.enabled** flag using the EE interceptor. When an application calls the **TimerService.getAllTimers()** method, JBoss EAP checks this flag. If this flag is set to **true**, JBoss EAP refreshes the Jakarta Enterprise Beans timers from database before returning the result.

In the previous JBoss EAP releases, the Jakarta Enterprise Beans timer reading could be refreshed in a database using the **refresh-interval** attribute found in **database-data-store**. Users could set the **refresh-interval** attribute value in milliseconds to refresh the Jakarta Enterprise Beans timer reading.

Access to runtime information from Jakarta Enterprise Beans

With JBoss EAP 7.4, you can access runtime data for Jakarta Enterprise Beans. Stateful session beans, stateless session beans, and singleton beans each return different runtime information. For example, the following command returns runtime data for a stateless session bean:

```
/deployment=ejb-management.jar/subsystem=ejb3/stateless-session-
bean=ManagedStatelessBean:read-resource(include-runtime)
```

Dynamic discovery of Jakarta Enterprise Beans over HTTP

With JBoss EAP 7.4, you can use dynamic discovery of Jakarta Enterprise Beans over HTTP. To use this capability, add a configuration similar to the following to the **ejb-remote** profile:

```
<remote connector-ref="http-remoting-connector" thread-pool-name="default">
    <channel-creation-options>
        <option name="MAX_OUTBOUND_MESSAGES" value="1234" type="remoting"/>
    </channel-creation-options>
    <profiles>
        <profile name="my-profile">
            <remote-http-connection name="ejb-http-connection" uri="http://127.0.0.1:8180/wildfly-
services"/>
        </profile>
    </profiles>
</remote>
```

Global configuration of compression for remote Jakarta Enterprise Beans calls

With JBoss EAP 7.4, you can configure compression of calls to remote Jakarta Enterprise Beans globally. To configure compression globally on a stand-alone client, specify the **default.compression** property in the **jboss-ejb-client.properties** file. To configure compression globally on a server, include the **default-compression** attribute in the **<client-conext>** element in the **jboss-ejb-client.xml** descriptor file in the application deployment unit.

```
<jboss-ejb-client xmlns="urn:jboss:ejb-client:1.4">
```

```
    <client-context default-compression="5">
        <profile name="example-profile" />
    </client-context>
</jboss-ejb-client>
```

1.7. HIBERNATE

Configuring the wildfly.jpa.skipquerydetach persistence unit property

You can configure the **wildfly.jpa.skipquerydetach** persistence unit property from the **persistence.xml** file of a container-managed persistence context.

The default value for **wildfly.jpa.skipquerydetach** is **false**. Use this setting to set a transaction-scoped persistence context to immediately detach query results from an open persistence context.

Configure **wildfly.jpa.skipquerydetach** as **true**, to set a transaction-scoped persistence context to detach query results when a persistence context is closed. This enables a non-standard specification extension.

For applications that have the non-standalone specification extension **jboss.as.jpa.deferdetach** set as **true**, you can also set **wildfly.jpa.skipquerydetach** as **true**.

1.8. WEB SERVICES

Integrating Elytron with web services clients

You can now configure web services clients to use Elytron automatically to obtain the credentials, the authentication method, and the SSL context.

If you use the web services client and assign configuration properties to it using the JBossWS API, you are not prompted for credentials or required to accept server certificates if the valid configuration is in the Elytron client. The following authentication methods are supported:

- Username Token Profile authentication
- HTTP Basic authentication
- TLS protocol

The configuration is specified by the **<webservices/>** element in **wildfly-config.xml**.

Ability for RESTEasy 3.x to access all standard MicroProfile ConfigSources

RESTEasy 3.x can now access all standard MicroProfile **ConfigSources**. The following additional **ConfigSources** are also added to RESTEasy 3.x:

- **servlet init-params** (ordinal 60)
- **filter init-params** (ordinal 50)
- **servlet context-params** (ordinal 40)

Previously, these capabilities were only included in RESTEasy 4.x. With this update, RESTEasy can access configuration parameters with or without the MicroProfile **ConfigSources**. In the absence of a MicroProfile Config implementation, RESTEasy falls back to the older method of gathering parameters from **ServletContext** parameters and **init** parameters.

Configuring SameSite cookie attribute

You can now configure the **SameSite** attribute for cookies in the current JBoss EAP release with a **samesite-cookie** predicated handler in the **undertow** subsystem. With this handler, you can update your server configuration without having to change your applications. This enhancement supports changes to the processing of cookies that were recently implemented in major web browsers to improve security.

1.9. MESSAGING

Ability to pause a topic

With JBoss EAP 7.4, you can pause a topic in addition to pausing a queue. When you pause a topic, JBoss EAP receives messages but does not deliver them. When you resume the topic, JBoss EAP delivers the messages. To pause a topic, issue a command similar to the following example:

```
/subsystem=messaging-activemq/server=default/jms-topic=topic:pause()
```

To resume a topic, issue a command similar to the following example:

```
/subsystem=messaging-activemq/server=default/jms-topic=topic:resume()
```

Artemis network health checks

With JBoss EAP 7.4, you can check the health of the network using the following Artemis network health check configuration parameters:

- **network-check-NIC**
- **network-check-period**
- **network-check-timeout**
- **network-check-list**
- **network-check-URL-list**
- **network-check-ping-command**
- **network-check-ping6-command**

For example, to check the network status by pinging the IP address **10.0.0.1**, issue the following command:

```
/subsystem=messaging-activemq/server=default:write-attribute(name=network-check-list,
value="10.0.0.1")
```

CHAPTER 2. UNSUPPORTED FUNCTIONALITY

2.1. UNSUPPORTED FEATURES

Support for some technologies are removed due to the high maintenance cost, low community interest, and better alternative solutions.

Platforms and features

JBoss EAP deprecated the following platforms in version 7.1. These platforms are not tested in JBoss EAP 7.4.

- Oracle Solaris on x86_64
- Oracle Solaris on SPARCv9

JBoss EAP 7.4 does not include the Wildfly SSL natives for these platforms. As a result, SSL operations in Oracle Solaris platforms might be slower than they were on previous versions of JBoss EAP.

RESTEasy Parameters

RESTEasy provides a Servlet 3.0 **ServletContainerInitializer** integration interface that performs an automatic scan of resources and providers for a servlet. Containers can use this integration interface to start an application. Therefore, use of the following RESTEasy parameters is no longer supported:

- resteasy.scan
- resteasy.scan.providers
- resteasy.scan.resources

Eclipse MicroProfile capabilities

The following Eclipse MicroProfile capabilities that were included as technical preview in JBoss EAP 7.3 are not included in JBoss EAP 7.4 beta:

- Eclipse MicroProfile Config
- Eclipse MicroProfile Health
- Eclipse MicroProfile Metrics
  JBoss EAP no longer includes the **microprofile-smallrye-metrics** subsystem, so application metrics endpoints are no longer available. JBoss EAP continues to include endpoints for JVM and server metrics.
- Eclipse MicroProfile OpenTracing
- Eclipse MicroProfile REST client

These capabilities are now part of the JBoss EAP Expansion Pack (JBoss EAP XP). Install the JBoss EAP XP for full Eclipse MicroProfile support in JBoss EAP. For complete information about support for Eclipse MicroProfile and JBoss EAP XP, see the JBoss EAP XP lifecycle and support policies page .

2.2. DEPRECATED FEATURES

Some features have been deprecated with this release. This means that no enhancements will be made to these features, and they may be removed in the future, usually the next major release.

Red Hat will continue providing full support and bug fixes under our standard support terms and conditions. For more information about the Red Hat support policy, see the Red Hat JBoss Middleware Product Update and Support Policy located on the Red Hat Customer Portal.

For details of which features have been deprecated, see the JBoss Enterprise Application Platform Component Details located on the Red Hat Customer Portal.

Platforms and features
Support for the following platforms and features is deprecated:

Operating systems

- Microsoft Windows Server on i686
- Red Hat Enterprise Linux (RHEL) 6 on i686

NOTE

Although support for these platforms was deprecated in a previous JBoss EAP release, some artifacts and resources linked to these platforms were not removed, such as the **wildfly-openssl** native library binding . For Red Hat JBoss Enterprise Application Platform 7.4, those artifacts and resources have been removed.

Databases and database connectors

- IBM DB2 11.1
- PostgreSQL / EnterpriseDB 11
- MariaDB 10.1
- MS SQL 2017

Lightweight Directory Access Protocol (LDAP) servers

- Red Hat Directory Server 10.0
- Red Hat Directory Server 10.1

Spring BOM
The following Spring BOM that is located in the Red Hat Maven repository is now deprecated:

- jboss-eap-jakartaee8-with-spring4

Although Red Hat tests that Spring applications run on Red Hat JBoss Enterprise Application Platform 7.4, you must use the latest version of the Spring Framework and its BOMs (for example, **x.y.z.RELEASE**) for developing your applications on JBoss EAP 7.4.

For more information about versions of the Spring Framework, see Spring Framework Versions on *GitHub*.

Java Development Kits (JDKs)

- JDK 8

JBoss EAP OpenShift templates
JBoss EAP templates for OpenShift are deprecated.

eap74-beta-starter-s2i.json and **eap73-third-party-db-s2i.json** templates
The **eap74-beta-starter-s2i.json** and **eap74-beta-third-party-db-s2i.json** templates are deprecated and will be removed in JBoss EAP 7.4.0.GA.

Legacy security subsystem
The **org.jboss.as.security** extension and the legacy **security** subsystem it supports are now deprecated. Migrate your security implementations from the **security** subsystem to the **elytron** subsystem.

PicketLink
The **org.wildfly.extension.picketlink** extension, and the **picketlink-federation** and **picketlink-identity-management** subsystems this extension supports, are now deprecated. Migrate your single sign-on implementation to Red Hat Single Sign-On.

Managed domain support for previous versions of JBoss EAP
Support for hosts running JBoss EAP 7.3 and earlier versions in a JBoss EAP 7.4 managed domain is deprecated. Migrate the hosts in your managed domains to JBoss EAP 7.4.

Server configuration files using namespaces from JBoss EAP 7.3 and earlier
Using server configuration files (**standalone.xml**, **host.xml**, and **domain.xml**) that include namespaces from JBoss EAP 7.3 and earlier is deprecated in this release. Update your server configuration files to use JBoss EAP 7.4 namespaces.

CHAPTER 3. RESOLVED ISSUES

See Resolved Issues for JBoss EAP 7.4 to view the list of issues that have been resolved for this release.

Additionally, be aware of the following:

- After completing source-to-image builds, OpenShift now clears the source directory (/**tmp**/**src**). As a result of this change, built images should be smaller.

CHAPTER 4. FIXED CVES

JBoss EAP 7.4 beta includes fixes for the following security-related issues:

- CVE-2019-14540: REST: jackson-databind: Polymorphic typing issue related to com.zaxxer.hikari.HikariConfig.
- CVE-2019-16942: REST: jackson-databind: Serialization gadgets in classes of the commons-dbcp package.
- CVE-2019-10086: Server: commons-beanutils: apache-commons-beanutils: does not suppress the class property in PropertyUtilsBean by default.
- CVE-2019-16943: REST: jackson-databind: Serialization gadgets in classes of the p6spy package.
- CVE-2019-20445: JMS: netty: **HttpObjectDecoder.java** allows the content-length header to be accompanied by a second content-length header.
- CVE-2019-17531: REST: jackson-databind: Polymorphic typing issue when enabling default typing for an externally exposed JSON endpoint and having **apache-log4j-extra** in the classpath leads to code execution.
- CVE-2019-20444: JMS: netty: HTTP request smuggling.
- CVE-2019-14888: Web (Undertow): undertow: Possible Denial Of Service (DOS) in Undertow HTTP server listening on HTTPS.
- CVE-2019-12423: Web Services: cxf-core: cxf: OpenId Connect token service does not properly validate the clientId.
- CVE-2019-14887: Management: wildfly: The **enabled-protocols** value in legacy security is not respected if OpenSSL security provider is in use.
- CVE-2019-0210: MP OpenTracing: libthrift: thrift: Out-of-bounds read related to **TJSONProtocol** or **TSimpleJSONProtocol**.
- CVE-2019-16869: JMS: netty: HTTP request smuggling by mishandled whitespace before the colon in HTTP headers.
- CVE-2020-1732: Security: wildfly: Soteria: Security identity corruption across concurrent threads.
- CVE-2020-7238: JMS: netty: HTTP request smuggling due to Transfer-Encoding whitespace mishandling.
- CVE-2020-1695: REST: resteasy-jaxrs: resteasy: Improper validation of response header in **MediaTypeHeaderDelegate.java** class.
- CVE-2019-14893: REST: jackson-databind: Serialization gadgets in classes of the xalan package.
- CVE-2019-16335: REST: jackson-databind: Polymorphic typing issue related to **com.zaxxer.hikari.HikariDataSource**.
- CVE-2019-14892: REST: jackson-databind: Serialization gadgets in classes of the commons-configuration package.

- CVE-2019-10174: Clustering: infinispan-core: infinispan: The **invokeAccessibly** method from the **ReflectionUtil** class allows to invoke private methods.
- CVE-2019-17267: REST: jackson-databind: Serialization gadgets in classes of the ehcache package.
- CVE-2020-10688: REST: resteasy: RESTEASY003870 exception in RESTEasy can lead to a reflected XSS attack.
- CVE-2019-12419: Web Services: xf-core: cxf: OpenId Connect token service does not properly validate the clientId.
- CVE-2020-1745: Web (Undertow): undertow: AJP File Read/Inclusion Vulnerability.
- CVE-2019-0205: MP OpenTracing: libthrift: thrift: Endless loop when fed with specific input data.
- CVE-2019-17573: Web Services: cxf: Reflected XSS in the services listing page.
- CVE-2020-10740: **wildfly**: Unsafe deserialization in Wildfly Enterprise Java Beans.
- CVE-2020-10714: **wildfly-elytron**: Session fixation when using FORM authentication.
- CVE-2020-6950 **Mojarra**: Path traversal using either the **loc** parameter or the **con** parameter, incomplete fix of CVE-2018-14371.
- CVE-2020-1954: **cxf-core**: **cxf**: JMX integration is vulnerable to a MITM attack.
- CVE-2018-14371: **jsf-impl**: **Mojarra**: Path traversal in **ResourceManager.java:getLocalePrefix()** using the **loc** parameter.
- CVE-2020-10683: **dom4j**: XML External Entity vulnerability in default SAX parser.
- CVE-2020-10705: **undertow**: Memory exhaustion issue in **HttpReadListener** with "Expect: 100-continue" header.
- CVE-2020-11612: **netty**: Compression/decompression codecs do not enforce limits on buffer allocation sizes.
- CVE-2020-1719: **wildfly**: The **EJBContext** principal is not popped back after invoking another EJB using a different security domain.
- CVE-2019-10172: **jackson-mapper-asl**: XML external entity similar to CVE-2016-3720.
- CVE-2020-10719: **undertow**: Invalid HTTP request with large chunk size.
- CVE-2020-10673: **jackson-databind**: The interaction between serialization gadgets and typing is mishandled and this could result in remote command execution.

CHAPTER 5. KNOWN ISSUES

See Known Issues for JBoss EAP 7.4 to view the list of known issues for this release.

Additionally, be aware of the following:

Setting OPENSHIFT_DNS_PING_SERVICE_NAME to an empty value results in boot error.

Do not set **OPENSHIFT_DNS_PING_SERVICE_NAME** to an empty value. A boot error occurs and clustering is disabled.

Revised on 2021-03-23 15:04:03 UTC