Open Liberty 2020

Release Notes for Open Liberty 20.0.0.7 on Red Hat OpenShift Container Platform

Release Notes for Open Liberty 2020 on Red Hat OpenShift Container Platform

Last Updated: 2020-07-06

Open Liberty 2020 Release Notes for Open Liberty 20.0.0.7 on Red Hat OpenShift Container Platform

Release Notes for Open Liberty 2020 on Red Hat OpenShift Container Platform

Legal Notice

Abstract

These release notes contain the latest information about new features, enhancements, fixes, and issues contained in Open Liberty 2020 on Red Hat OpenShift Container Platform release.

Table of Contents

CHAPTER 1. FEATURES

With Open Liberty 20.0.0.7, you can now disable the default LTPA cookies being returned during authentication when using TAI or SPNEGO authentication, and disable JWT cookies being returned when using the JWT Single Sign-on feature.

In Open Liberty 20.0.0.7:

- Choose to disable LTPA cookies for TAI or SPNEGO
- Choose to disable JWT cookies
- Significant bugs fixed in this release

1.1. RUN YOUR APPS USING 20.0.0.7

If you're using Maven, here are the coordinates:

```
<dependency>
    <groupId>io.openliberty</groupId>
    <artifactId>openliberty-runtime</artifactId>
    <version>20.0.0.7</version>
    <type>zip</type>
</dependency>
```

Or for Gradle:

```
dependencies {
    libertyRuntime group: 'io.openliberty', name: 'openliberty-runtime', version: '[20.0.0.7,)'
}
```

Or if you're using Docker:

```
FROM open-liberty
```

Or take a look at our Downloads page, where we now also have the Kernel package available to download as a ZIP file. You can then use the featureUtility command to add the features that you need to the kernel.

1.1.1. Choose to disable LTPA cookies for TAI or SPNEGO

LTPA cookies contain an encrypted authentication token with user identity and expiration information and can be used for single sign-on (SSO). You can now decide whether to receive the LTPA cookie when using TAI and SPNEGO authentication.

When a client (like a browser) is authenticated with an Open Liberty server, the default response is to receive an SSO LTPA cookie in the HTTP servlet. When the same client accesses another protected resource in the Open Liberty server that shares the same LTPA configuration, authentication with the SSO LTPA cookie happens first, before any other authentication mechanism. This can potentially cause unintended results if another authentication mechanism is to be used. You can now disable the creation of LTPA cookies when using TAI and SPNEGO authentication.

Disable LTPA cookies for TAI in the **server.xml**:

```
<server>
  <featureManager>
    <feature>appSecurity-2.0</feature>
  </featureManager>
  <trustAssociation id="sample" disableLtpaCookie="true" />
</server>
```

Disable LTPA cookies for SPNEGO in the **server.xml**:

```
<server>
  <featureManager>
    <feature>spnego-1.0</feature>
  </featureManager>
  <spnego id="sample" disableLtpaCookie="true" />
</server>
```

1.1.2. Choose to disable JWT SSO cookie

When a client (like a browser) is authenticated with an Open Liberty server through the JSON Web Tokens (JWT) single sign-on (SSO) feature (**jwtSso-1.0**), the default response is a JWT SSO cookie in the HTTP servlet. When the same client accesses another protected resource in the same or in a different Open Liberty server, authentication with the JWT cookie happens first, before any other authentication mechanism. This can potentially cause unintended results if another authentication mechanism is to be used. You can now disable JWT cookies when using the JWT SSO feature.

Disable JWT cookies for JWT SSO in the **server.xml**:

```
<server>
  <featureManager>
    <feature>jwtSso-1.0</feature>
  </featureManager>
  <jwtSso id="sample" disableJwtCookie="true" />
</server>
```

1.2. SIGNIFICANT BUGS FIXED IN THIS RELEASE

We've spent some time fixing bugs. The following sections describe just some of the issues we resolved in this release. If you're interested, here's the full list of fixed bugs in 20.0.0.7 .

1.2.1. Notable bug fixes and enhancements in JAX-RS 2.1

If you've been seeing a **NullPointerException** when writing multipart form data in your JAX-RS response, we've got good news for you - we fixed that in issue 8048!

One of our users needed a clever way to restrict JSON field serialization by a user's security role. By using a **ContextResolver** for specifying the JSON-B visibility strategy and injecting the **SecurityContext**, they were able to make this work. Only one problem - the injection into the ContextResolver wasn't working... We fixed that too! Check out issue 12715. It's a pretty cool use case!

1.2.2. Improvements to HTTP/2 Implementation

A scenario was reported where excess CPU consumption is seen when a client does not terminate a HTTP/2 connection gracefully. We've resolved this in issue 12599.

In some specific cases, Liberty does not update its HTTP/2 read window quickly enough, causing the flow control window to stall. We have improved Liberty’s flow control behavior with 12399

CHAPTER 2. RESOLVED ISSUES

See the Open Liberty 20.0.0.7 issues that were resolved for this release .

CHAPTER 3. FIXED CVES

For a list of CVEs that were fixed in Open Liberty 20.0.0.7, see security vulnerabilities.

CHAPTER 4. KNOWN ISSUES

See the list of issues that were found but not fixed during the development of 20.0.0.7 .