



Open Liberty 2020

Release Notes for Open Liberty 20.0.0.8 on Red Hat OpenShift Container Platform

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

Open Liberty 2020 Release Notes for Open Liberty 20.0.0.8 on Red Hat OpenShift Container Platform

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

Legal Notice

Copyright © 2020 IBM Corp

Code and build scripts are licensed under the Eclipse Public License v1 Documentation files are licensed under Creative Commons Attribution-NoDerivatives 4.0 International (CC BY-ND 4.0)

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	3
1.1. RUN YOUR APPS USING 20.0.0.8	3
1.1.1. HTTP access log fields in JSON logs	3
1.1.2. Allow JSON written to System.out to pass without wrapping	5
CHAPTER 2. RESOLVED ISSUES	7
CHAPTER 3. FIXED CVES	8
CHAPTER 4. KNOWN ISSUES	9

CHAPTER 1. FEATURES

With Open Liberty 20.0.0.8 you can now access HTTP log fields in JSON logs which allows you to include fields from the **accessLogging logFormat** attribute in your JSON logs. The update also allows for JSON to be written directly to **system.out** without wrapping in the **liberty_message** event.

In [Open Liberty 20.0.0.8](#):

- [HTTP access log fields in JSON logs](#)
- [Allow JSON written to System.out to pass without wrapping](#)

View the list of fixed bugs in [20.0.0.8](#).

1.1. RUN YOUR APPS USING 20.0.0.8

If you're using [Maven](#), here are the coordinates:

```
<dependency>
  <groupId>io.openliberty</groupId>
  <artifactId>openliberty-runtime</artifactId>
  <version>20.0.0.8</version>
  <type>zip</type>
</dependency>
```

Or for [Gradle](#):

```
dependencies {
  libertyRuntime group: 'io.openliberty', name: 'openliberty-runtime', version: '[20.0.0.8,)'
}
```

Or if you're using Docker:

```
FROM open-liberty
```

1.1.1. HTTP access log fields in JSON logs

In Open Liberty, you have the option to format your server logs in either basic or JSON format. When logs are in JSON format, you have to specify the sources (**message**, **trace**, **accessLog**, **ffdc**, **audit**) you want to send to **messages.log** or **console.log** and **standard-out**. Open Liberty 20.0.0.8 adds the option for you to include fields from the **accessLogging logFormat** attribute in the JSON logs. Previously, only selected fields would be printed in JSON logs. You are now able to include other NCSA (National Center for Supercomputing Applications) access log fields in your JSON logs.

Also, you now have the ability to define which JSON access log fields you want from the **accessLogging logFormat** attribute, which can then be sent to a log analysis tool, such as the ELK (Elasticsearch, Logstash, Kibana) stack. With this new feature, you can receive more informative logs that suit your needs. For example, you are able to specify that you want the user ID and request time fields in your JSON access logs – then, these can be filtered by user ID in Kibana and track performance on a user-by-user basis. When logs are in JSON format, you can use the new **jsonAccessLogFields** logging attribute to specify whether you want your access logs to have the default set of fields or a custom set of fields based on the HTTP **accessLogging logFormat** attribute. In order to receive access logs, the property **accessLogging** or **httpAccessLogging** must be set. For example, you can set the following attributes in your **server.xml**:

```
<httpEndpoint id="defaultHttpEndpoint" httpPort="9080" httpsPort="9443" host="">
  <accessLogging logFormat='%R{W} %u %{my_cookie}C %s'/>
</httpEndpoint>
<logging messageFormat="json" messageSource="message,accessLog"
  jsonAccessLogFields="logFormat"/>
```

Now, in the **messages.log** file, your access logs will contain the four fields specified in the **accessLogging logFormat** attribute (elapsed time, user ID, cookie, and response code).

```
{
  "type": "liberty_accesslog",
  "host": "192.168.1.15",
  "ibm_userDir": "/you/jennifer.zhen.chengibm.com/libertyGit/open-liberty/dev/build.image/wlp/usr/",
  "ibm_serverName": "defaultServer",
  "ibm_cookie_my_cookie": "example_cookie",
  "ibm_responseCode": 200,
  "ibm_datetime": "2020-06-18T09:30:47.693-0400",
  "ibm_sequence": "1592487047653_00000000000001"
}
```

This new functionality is also available for the logstashCollector-1.0 feature by adding the following to your **server.xml**:

```
<featureManager>
  <feature>logstashCollector-1.0</feature>
</featureManager>

<logstashCollector
  jsonAccessLogFields="logFormat">
</logstashCollector>
```

This table describes the new fields available with the corresponding **logFormat** token:

Field	Description	logFormat Token
ibm_remotelIP	Remote IP address, e.g. 127.0.0.1	%a
ibm_bytesSent	Response size in bytes excluding headers.	%b
ibm_cookie_{cookienam e}	Cookie value from the request.	%{cookieName}C or %C
ibm_requestElapsedTime	The elapsed time of the request - millisecond accuracy, microsecond precision.	%D
ibm_requestHeader_{headerna me}	Header value from the request.	%{headerName}i
ibm_responseHeader_{headerna me}	Header value from the response.	%{headerName}o

Field	Description	logFormat Token
ibm_requestFirstLine	First line of the request.	%r
ibm_requestStartTime	The start time of the request, in NCSA format.	%t
ibm_accessLogDatetime	The time when the message to the access log is queued to be logged, in normal NCSA format.	%{t}W
ibm_remoteUserID	Remote user according to the WebSphere Application Server specific \$WSRU header	%u

For more information:

- [Open Liberty Logging](#)
- [Open Liberty Log and Trace Configuration](#)

1.1.2. Allow JSON written to System.out to pass without wrapping

In Open Liberty, you have the option to format your server logs in basic or JSON format. When logs are in JSON format, you have to specify the sources (message, trace, accessLog, ffdc, audit) you want to send to **messages.log** or **console.log** and **standard.out**. This feature adds a new option for you to write pre-formatted JSON application logs directly to **System.out** or **System.err**. Previously, when Open Liberty was running with JSON logging enabled, Open Liberty embeds anything written to **System.out** or **System.err** into the message field of a **liberty_message** event. Now, you can write JSON directly to **System.out** or **System.err** without wrapping in the **liberty_message** event.

You can now write your JSON application logs directly to **System.out** or **System.err**, which can then be sent to a log analysis tool, such as the ELK (Elasticsearch, Logstash, Kibana) stack. Previously, when JSON logging was enabled, pre-formatted JSON application logs would look like this:

```
{
  "type":"liberty_message",
  "host":"192.168.0.119",
  "ibm_userDir":"\\you\\yushan.lin@ibm.com\\Documents\\archived-guide-
log4j\\finish\\target\\liberty\\wlp\\usr\\
",
  "ibm_serverName":"log4j.sampleServer",
  "message":{"\n  \"timeMillis\" : 1587666082123,\n
    \"thread\" : \"Default Executor-thread-8\",\n
    \"level\" : \"WARN\",\n
    \"loggerName\" : \"application.servlet.LibertyServlet\",\n
    \"message\" : \"hello liberty servlet warning message!\",\n
    \"endOfBatch\" : false,\n
    \"loggerFqcn\" : \"org.apache.logging.log4j.spi.AbstractLogger\",\n
    \"threadId\" : 53,\n
    \"threadPriority\" : 5\n}\r",
  "ibm_threadId":"00000035",
```

```
"ibm_datetime":"2020-04-23T14:21:22.124-0400",  
"module":"SystemOut",  
"loglevel":"SystemOut",  
"ibm_methodName": "",  
"ibm_className": "",  
"ibm_sequence":"1587666082124_000000000001B",  
"ext_thread":"Default Executor-thread-8"  
}
```

Now you can output the logs so that pre-formatted JSON logs are not wrapped in **liberty_message** events. Visualization tools such as Kibana can be used to analyze certain fields in your JSON logs. You can analyze both custom-formatted JSON application logs and Liberty JSON logs in the same visualization.

You can enable this functionality by setting **appsWriteJson="true"** in the logging element of the **server.xml**, or you can have it set from the moment the server starts by setting in bootstrap.properties: **com.ibm.ws.logging.apps.write.json=true**.

For more information:

- [Open Liberty Logging](#)

CHAPTER 2. RESOLVED ISSUES

See the [Open Liberty 20.0.0.8 issues that were resolved for this release](#) .

CHAPTER 3. FIXED CVEs

For a list of CVEs that were fixed in Open Liberty 20.0.0.8, see [security vulnerabilities](#).

CHAPTER 4. KNOWN ISSUES

See the [list of issues that were found but not fixed during the development of 20.0.0.8](#) .