Red Hat build of OpenJDK 21

Release notes for Eclipse Temurin 21.0.8

Last Updated: 2025-07-24

Legal Notice

Abstract

The release notes for Eclipse Temurin 21.0.8 provide an overview of new features in OpenJDK 21 and a list of potential known issues and possible workarounds.

Table of Contents

PREFACE

Open Java Development Kit (OpenJDK) is a free and open source implementation of the Java Platform, Standard Edition (Java SE). Eclipse Temurin is available in four LTS versions: OpenJDK 8u, OpenJDK 11u, OpenJDK 17u, and OpenJDK 21u.

Binary files for Eclipse Temurin are available for macOS, Microsoft Windows, and multiple Linux x86 Operating Systems including Red Hat Enterprise Linux and Ubuntu.

PROVIDING FEEDBACK ON RED HAT BUILD OF OPENJDK DOCUMENTATION

To report an error or to improve our documentation, log in to your Red Hat Jira account and submit an issue. If you do not have a Red Hat Jira account, then you will be prompted to create an account.

Procedure

1. Click the following link to **create a ticket**.
2. Enter a brief description of the issue in the **Summary**.
3. Provide a detailed description of the issue or enhancement in the **Description**. Include a URL to where the issue occurs in the documentation.
4. Clicking **Create** creates and routes the issue to the appropriate documentation team.

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

CHAPTER 1. SUPPORT POLICY FOR ECLIPSE TEMURIN

Red Hat will support select major versions of Eclipse Temurin in its products. For consistency, these versions remain similar to Oracle JDK versions that Oracle designates as long-term support (LTS).

A major version of Eclipse Temurin will be supported for a minimum of six years from the time that version is first introduced. For more information, see the Eclipse Temurin Life Cycle and Support Policy .

NOTE

RHEL 6 reached the end of life in November 2020. Because of this, Eclipse Temurin does not support RHEL 6 as a supported configuration.

CHAPTER 2. ECLIPSE TEMURIN FEATURES

Eclipse Temurin does not contain structural changes from the upstream distribution of OpenJDK.

For more information about the changes and security fixes in the latest OpenJDK 21 release of Eclipse Temurin, see OpenJDK 21.0.8 Released.

2.1. NEW FEATURES AND ENHANCEMENTS

Eclipse Temurin 21.0.8 includes the following new features and enhancements.

Fix for type annotations remaining invisible to javac plugins across compilation boundaries

In earlier OpenJDK releases, the **javac** compiler provided access to type annotations only when types were loaded from source code files. If a type was loaded from bytecode, any type annotations were omitted.

In OpenJDK 21.0.8, the **TypeMirror** interface also provides access to annotations for types that are loaded from bytecode. You can obtain these annotations by using the **AnnotationMirror#getAnnotationMirrors** method. These annotations are included in the output of the **AnnotationMirror#toString** method.

If your programs rely on type annotations being absent from elements that are loaded from bytecode, you must update these programs to handle type annotations.

Because of ongoing issues (see JDK-8360406), this feature is not enabled by default. If you want the **javac** compiler to include type annotations for types that are loaded from bytecode, specify the **-XDaddTypeAnnotationsToSymbol=true** option.

See JDK-8341779 (JDK Bug System).

Improved HTTP/2 flow control checks

OpenJDK 21.0.8 enhances the HTTP/2 client implementation in **java.net.http.HttpClient** objects to report flow control errors to the server. This behavior is typically transparent in most situations. However, it might mean that streams are reset or connections are closed when connecting to an HTTP/2 server that cannot correctly handle these errors.

You can use the following existing properties to adjust flow control limits:

jdk.httpclient.connectionWindowSize

- Specifies the HTTP/2 client connection window size in bytes
- Default value: **2^26**
- Range: **2^16-1** to **2^31-1**

jdk.httpclient.windowSize

- Specifies the HTTP/2 client stream window size in bytes
- Default value: **16777216** (16 MB)
- Range: **2^14** to **2^31-1**

If you specify an invalid value, the default value is used. This enhancement guarantees that the actual value for the connection window size is not smaller than the stream window size.

See JDK-8342075 (JDK Bug System).

New diagnostic commands for jcmd to print annotated process memory map

In OpenJDK 21.0.8, the **jcmd** utility includes the following new commands for printing the virtual memory map of the JVM to either standard output (**stdout**) or a file.

jcmd *<pid>* System.map

Prints the virtual memory map of the JVM identified by ***<pid>*** to **stdout**

jcmd *<pid>* System.dump_map

Prints the virtual memory map of the JVM identified by ***<pid>*** to a **vm_memory_map_<pid>.txt** file in the current directory

If native memory tracking (NMT) is enabled, these commands also print NMT information about the virtual memory segments.

See JDK-8318636 (JDK Bug System).

Updated HSS/LMS public key encoding

In OpenJDK 21.0.8, the X.509 encoding format for hierarchical signature system (HSS) and Leighton-Micali signature (LMS) public keys now aligns with the latest standard in the Internet Engineering Task Force (IETF) RFC 9708. This enhancement also means that the **OCTET_STRING** wrapping around the public key value has been removed.

For compatibility with earlier releases, the JDK still detects the presence of distinguished encoding rules (DER) encoding when reading keys that were encoded before this release.

See JDK-8347596 (JDK Bug System).

Compatibility of OCSP readtimeout property with OCSP timeout

The initial release of OpenJDK 21 introduced the **com.sun.security.ocsp.readtimeout** property, which specifies the timeout for reading Online Certificate Status Protocol (OCSP) data. This property was paired with the existing **com.sun.security.ocsp.timeout** property to provide greater control over the timeouts for OCSP connections and certificate retrieval. By using these properties, you can set timeouts for reading OCSP data and for the transport layer independently of each other.

In earlier OpenJDK 21 releases, if you did not specify a value for **com.sun.security.ocsp.readtimeout**, the default timeout for reading OCSP data was 15 seconds.

In OpenJDK 21.0.8, if you do not specify a value for **com.sun.security.ocsp.readtimeout**, the default timeout for reading OCSP data is based on the value of the **com.sun.security.ocsp.timeout** property. If you similarly do not specify a value for **com.sun.security.ocsp.timeout**, the default timeout is 15 seconds, as in earlier releases.

NOTE

This enhancement matches the **com.sun.security.ocsp.readtimeout** behavior that was introduced in OpenJDK 17.0.15. This enhancement also matches the default behavior that existed in OpenJDK versions before the **com.sun.security.ocsp.readtimeout** property was introduced.

See JDK-8347506 (JDK Bug System).

Sectigo CS and TLS root certificates added

In OpenJDK 21.0.8, the **cacerts** truststore includes four Sectigo root certificates, including two code-signing (CS) certificates and two TLS certificates:

Certificate 1

- Name: Sectigo Limited
- Alias name: sectigocodesignroote46
- Distinguished name: CN=Sectigo Public Code Signing Root E46, O=Sectigo Limited, C=GB

Certificate 2

- Name: Sectigo Limited
- Alias name: sectigocodesignrootr46
- Distinguished name: CN=Sectigo Public Code Signing Root R46, O=Sectigo Limited, C=GB

Certificate 3

- Name: Sectigo Limited
- Alias name: sectigotlsroote46
- Distinguished name: CN=Sectigo Public Server Authentication Root E46, O=Sectigo Limited, C=GB

Certificate 4

- Name: Sectigo Limited
- Alias name: sectigotlsrootr46
- Distinguished name: CN=Sectigo Public Server Authentication Root R46, O=Sectigo Limited, C=GB

See JDK-8359170 (JDK Bug System).

Fix for potential failures in region allocation when using the G1 garbage collector with multiple NUMA nodes

On non-uniform memory access (NUMA) systems, the operating system can choose to migrate a task from one NUMA node to another. In the Garbage-First (G1) garbage collector, **G1AllocRegion** objects are associated with NUMA nodes. The **G1Allocator** code obtains the **G1AllocRegion** object for the current thread only, but operating system scheduling can lead to arbitrary changes in the NUMA-to-thread association.

In earlier OpenJDK releases, when using the G1 garbage collector with multiple NUMA nodes, a failure might have resulted when the **G1AllocRegion** object being used was changed mid-operation.

OpenJDK 21.0.8 resolves this issue by ensuring that the same NUMA node and associated **G1AllocRegion** object are used throughout an operation.

See JDK-8351500 (JDK Bug System)

2.2. DEPRECATED FEATURES

The following pre-existing features have been either deprecated or removed in Eclipse Temurin 21.0.8:

Baltimore root certificate removed

From OpenJDK 21.0.8 onward, the **cacerts** truststore no longer includes the following Baltimore root certificate that expired in May 2025:

- Alias name: baltimorecybertrustca [jdk]
- Distinguished name: CN=Baltimore CyberTrust Root, OU=CyberTrust, O=Baltimore, C=IE

See JDK-8303770 (JDK Bug System).

Camerfirma root CA certificates removed

From OpenJDK 21.0.8 onward, the **cacerts** truststore no longer includes the following expired Camerfirma root certificates:

Certificate 1

- Alias name: camerfirmachamberscommerceca [jdk]
- Distinguished name: CN=Chambers of Commerce Root OU=http://www.chambersign.org O=AC Camerfirma SA CIF A82743287 C=EU
- SHA256: 0C:25:8A:12:A5:67:4A:EF:25:F2:8B:A7:DC:FA:EC:EE:A3:48:E5:41:E6:F5:CC:4E:E6:3B:71:B3:61:6C

Certificate 2

- Alias name: camerfirmachambersignca [jdk]
- Distinguished name: CN=Global Chambersign Root - 2008 O=AC Camerfirma S.A. SERIALNUMBER=A82743287 L=Madrid (see current address at www.camerfirma.com/address) C=EU
- SHA256: 13:63:35:43:93:34:A7:69:80:16:A0:D3:24:DE:72:28:4E:07:9D:7B:52:20:BB:8F:BD:74:78:16:EE:BE

See JDK-8350498 (JDK Bug System).

Revised on 2025-07-24 17:52:53 UTC