Red Hat build of OpenJDK 11

Release notes for Red Hat build of OpenJDK 11.0.21

Last Updated: 2023-10-24

Red Hat build of OpenJDK 11 Release notes for Red Hat build of OpenJDK 11.0.21

Legal Notice

Abstract

The Release notes for Red Hat build of OpenJDK 11.0.21 document provides an overview of new features in Red Hat build of OpenJDK 11 and a list of potential known issues and possible workarounds.

Table of Contents

PREFACE

OpenJDK (Open Java Development Kit) is a free and open source implementation of the Java Platform, Standard Edition (Java SE). The Red Hat build of OpenJDK is available in three versions: 8u, 11u, and 17u.

Packages for the Red Hat build of OpenJDK are made available on Red Hat Enterprise Linux and Microsoft Windows and shipped as a JDK and JRE in the Red Hat Ecosystem Catalog.

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.

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your feedback on our documentation. To provide feedback, you can highlight the text in a document and add comments.

This section explains how to submit feedback.

Prerequisites

- You are logged in to the Red Hat Customer Portal.
- In the Red Hat Customer Portal, view the document in **Multi-page HTML** format.

Procedure

To provide your feedback, perform the following steps:

1. Click the **Feedback** button in the top-right corner of the document to see existing feedback.

NOTE

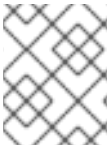
The feedback feature is enabled only in the **Multi-page HTML** format.

2. Highlight the section of the document where you want to provide feedback.
3. Click the **Add Feedback** pop-up that appears near the highlighted text.
   A text box appears in the feedback section on the right side of the page.
4. Enter your feedback in the text box and click **Submit**.
   A documentation issue is created.
5. To view the issue, click the issue tracker link in the feedback view.

CHAPTER 1. SUPPORT POLICY FOR RED HAT BUILD OF OPENJDK

Red Hat will support select major versions of Red Hat build of OpenJDK in its products. For consistency, these are the same versions that Oracle designates as long-term support (LTS) for the Oracle JDK.

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



NOTE

RHEL 6 reached the end of life in November 2020. Because of this, Red Hat build of OpenJDK is not supporting RHEL 6 as a supported configuration.

CHAPTER 2. DIFFERENCES FROM UPSTREAM OPENJDK 11

Red Hat build of OpenJDK in Red Hat Enterprise Linux (RHEL) contains a number of structural changes from the upstream distribution of OpenJDK. The Microsoft Windows version of Red Hat build of OpenJDK attempts to follow RHEL updates as closely as possible.

The following list details the most notable Red Hat build of OpenJDK 11 changes:

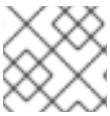
- FIPS support. Red Hat build of OpenJDK 11 automatically detects whether RHEL is in FIPS mode and automatically configures Red Hat build of OpenJDK 11 to operate in that mode. This change does not apply to Red Hat build of OpenJDK builds for Microsoft Windows.
- Cryptographic policy support. Red Hat build of OpenJDK 11 obtains the list of enabled cryptographic algorithms and key size constraints from RHEL. These configuration components are used by the Transport Layer Security (TLS) encryption protocol, the certificate path validation, and any signed JARs. You can set different security profiles to balance safety and compatibility. This change does not apply to Red Hat build of OpenJDK builds for Microsoft Windows.
- Red Hat build of OpenJDK on RHEL dynamically links against native libraries such as **zlib** for archive format support and **libjpeg-turbo**, **libpng**, and **giflib** for image support. RHEL also dynamically links against **Harfbuzz** and **Freetype** for font rendering and management.
- The **src.zip** file includes the source for all the JAR libraries shipped with Red Hat build of OpenJDK.
- Red Hat build of OpenJDK on RHEL uses system-wide timezone data files as a source for timezone information.
- Red Hat build of OpenJDK on RHEL uses system-wide CA certificates.
- Red Hat build of OpenJDK on Microsoft Windows includes the latest available timezone data from RHEL.
- Red Hat build of OpenJDK on Microsoft Windows uses the latest available CA certificate from RHEL.

Additional resources

- For more information about detecting if a system is in FIPS mode, see the Improve system FIPS detection example on the Red Hat RHEL Planning Jira.
- For more information about cryptographic policies, see Using system-wide cryptographic policies.

CHAPTER 3. RED HAT BUILD OF OPENJDK FEATURES

The latest Red Hat build of OpenJDK 11 release might include new features. Additionally, the latest release might enhance, deprecate, or remove features that originated from previous Red Hat build of OpenJDK 11 releases.



NOTE

For all the other changes and security fixes, see OpenJDK 11.0.21 Released.

3.1. RED HAT BUILD OF OPENJDK NEW FEATURES AND ENHANCEMENTS

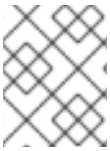
Review the following release notes to understand new features and feature enhancements that Red Hat build of OpenJDK 11.0.21 provides:

Increased default group size of TLS Diffie-Hellman

In Red Hat build of OpenJDK 11.0.21, the JDK implementation of Transport Layer Security (TLS) 1.2 uses a default Diffie-Hellman key size of 2048 bits. This supersedes the behavior in previous releases where the default Diffie-Hellman key size was 1024 bits.

This enhancement is relevant when a **TLS_DHE** cipher suite is negotiated and either the client or the server does not support Finite Field Diffie-Hellman Ephemeral (FFDHE) parameters. The JDK TLS implementation supports FFDHE, which is enabled by default and can negotiate a stronger key size.

As a workaround, you can revert to the previous key size by setting the **jdk.tls.ephemeralDHKeySize** system property to **1024**. However, to mitigate risk, consider using the default key size of 2048 bits.



NOTE

This change does not affect TLS 1.3, which already uses a minimum Diffie-Hellman key size of 2048 bits.

See JDK-8301700 (JDK Bug System).

Server-side cipher suite preferences used by default

In Red Hat build of OpenJDK 11.0.21, the SunJSSE provider uses the local server-side cipher suite preferences by default. This supersedes the behavior in previous releases where the server used the preferences that the connecting client specified.

You can revert to the previous behavior by using **SSLParameters.setUseCipherSuitesOrder(false)** on the server side.

See JDK-8168261 (JDK Bug System).

Support for RSA keys in PKCS#1 format

JDK providers can now accept Rivest–Shamir–Adleman (RSA) private and public keys in PKCS#1 format, such as the RSA **KeyFactory.impl** from the SunRsaSign provider. This feature requires that the RSA private or public key object has a PKCS#1 format and an encoding that matches the ASN.1 syntax for a PKCS#1 RSA private key and public key.

See JDK-8023980 (JDK Bug System).

-XshowSettings:locale output includes tzdata version

In Red Hat build of OpenJDK 11.0.21, the **-XshowSettings** launcher option also prints the **tzdata** version that the JDK uses. The **tzdata** version is displayed as part of the output for the **-XshowSettings:locale** option.

For example:

```
Locale settings:
    default locale = English
    default display locale = English
    default format locale = English
    tzdata version = 2023c
```

See JDK-8305950 (JDK Bug System).

Certigna root CA certificate added

In Red Hat build of OpenJDK 11.0.21, the **cacerts** truststore includes the Certigna root certificate:

- Name: Certigna (Dhimyotis)
- Alias name: certignarootca
- Distinguished name: CN=Certigna Root CA, OU=0002 48146308100036, O=Dhimyotis, C=FR

See JDK-8314960 (JDK Bug System).

Error thrown if default java.security file fails to load

In previous releases, if the **java.security** file failed to load successfully, Red Hat build of OpenJDK used a hardcoded set of security properties. However, this set of properties was poorly maintained and it was not obvious to users that the JDK was using these utilities.

To address this issue, if the **java.security** file fails to load successfully, Red Hat build of OpenJDK 11.0.21 throws an **InternalError** instead.

See JDK-8155246 (JDK Bug System).

Arrays cloned in several JAAS callback classes

In previous releases, in the **ChoiceCallback** and **ConfirmationCallback** JAAS classes, when arrays were passed into a constructor or returned, these arrays were not cloned. This behavior allowed an external program to gain access to the internal fields of these classes.

In Red Hat build of OpenJDK 11.0.21, the JAAS classes return cloned arrays.

See JDK-8242330 (JDK Bug System).

3.2. RED HAT BUILD OF OPENJDK DEPRECATED FEATURES

Review the following release notes to understand pre-existing features that have been either deprecated or removed in Red Hat build of OpenJDK 11.0.21:

SECOM Trust Systems root CA1 certificate removed

From Red Hat build of OpenJDK 11.0.21 onward, the **cacerts** truststore no longer includes the SECOM Trust Systems root certificate:

- Alias name: secomscrootca1 [jdk]
- Distinguished name: OU=Security Communication RootCA1, O=SECOM Trust.net, C=JP

See JDK-8295894 (JDK Bug System).

CHAPTER 4. ADVISORIES RELATED TO THIS RELEASE

The following advisories are issued to document bug fixes and CVE fixes included in this release:

- RHSA-2023:5734
- RHSA-2023:5735
- RHSA-2023:5736
- RHSA-2023:5737
- RHSA-2023:5739
- RHSA-2023:5740
- RHSA-2023:5741
- RHSA-2023:5742
- RHSA-2023:5743
- RHSA-2023:5744

Revised on 2023-10-24 15:23:53 UTC