OpenJDK 11

OpenJDK 11.0.9 リリースノート

Last Updated: 2020-11-26

法律上の通知

概要

本書では、OpenJDK 11 の新機能と概要と、潜在的な既知の問題および回避策を説明します。

目次

はじめに

OpenJDK (Open Java Development Kit) は、Java Platform Standard Edition (Java SE) のオープンソース実装です。OpenJDK の Red Hat ビルドは、OpenJDK 8u と OpenJDK 11u の 2 つのバージョンで利用できます。

Red Hat ビルドの OpenJDK 向けパッケージは、Red Hat Enterprise Linux および Microsoft Windows で利用でき、Red Hat Container Catalog の JDK および JRE として同梱されています。

第1章 OPENJDK のサポートポリシー

Red Hat は、一部の OpenJDK のメジャーバージョンをサポートします。一貫性のため、これらのバージョンは Oracle JDK 向けに「LTS」を規定するバージョンと同じです。

OpenJDK のメジャーバージョンは、最初に導入された時点から最低 6 年間サポートされます。

OpenJDK 11 の Microsoft Windows および Red Hat Enterprise Linux サポートは 2024 年 10 月までです。

注記

RHEL 6 のライフサイクルは 2020 年 11 月に終了します。このため、OpenJDK は 2020 年 10 月リリース以降のサポート対象設定として RHEL 6 をサポートしません。

詳細は、「OpenJDK のライフサイクルおよびサポートポリシー」を参照してください。

第2章 アップストリームの OPENJDK 11 との相違点

Red Hat Enterprise Linux の OpenJDK には、OpenJDK のアップストリームディストリビューションの構造上の変更が数多く含まれています。OpenJDK の Windows バージョンは、可能な限り Red Hat Enterprise Linux を追跡しようとします。

主な変更点は以下のとおりです。

- Red Hat Enterprise Linux では、アーカイブ形式のサポート (**zlib**) およびイメージ形式 (**libjpeg-turbo**、**libpng**、および **giflib**) に外部ネイティブライブラリーが使用されます。
  Microsoft Windows では、これらのライブラリーは、対応する Red Hat Enterprise Linux RPM のソースから構築され、動的リンクライブラリー (DLL) としてパッケージ化されます。

- Red Hat Enterprise inux では、システム全体のタイムゾーンデータファイルは、タイムゾーン情報のソースとして使用されます。
  Microsoft Windows には、Red Hat Enterprise Linux の利用可能な最新のタイムゾーンデータが含まれています。

- Red Hat Enterprise Linux では、システム全体の CA 証明書が使用されます。
  Microsoft Windows では、Red Hat Enterprise Linux からの利用可能な最新の CA 証明書が使用されます。

- **src.zip** ファイルには、OpenJDK に同梱されるすべての JAR ライブラリーのソースが含まれます。

第3章 OPENJDK の機能

3.1. 新機能および改良された機能

本項では、本リリースで導入された新機能を説明します。また、既存の機能の変更に関する情報も含まれます。

注記

その他の変更点やセキュリティー修正については、https://mail.openjdk.java.net/pipermail/jdk-updates-dev/2020-October/004007.html を参照してください。

3.1.1. MS950 文字セットレコーダーの変換テーブルが変更されました。

一方向のバイトツー文字マッピングの一部が、Unicode Consortium が提供する優先マッピングに合わせて調整されています。

詳細は、JDK-8240196 を参照してください。

3.1.2. 外部 FIPS モジュールがセキュリティーモジュールデータベースに存在する場合に、NSS で SunPKCS11 の初期化を許可する

FIPS 対応外部モジュールが Security Modules Database (NSSDB) で設定されている場合、SunPKCS11 セキュリティープロバイダーを NSS で初期化できるようになりました。この変更以前は、SunPKCS11 プロバイダーは、FIPS 以外のモードで NSS に対して設定されている場合に「FIPS flag set for non-internal module」というメッセージで RuntimeException をスローしました。

この変更により、システム全体の FIPS ポリシーが有効な場合に、OpenJDK は GNU/Linux オペレーティングシステムの最新の NSS リリースで適切に機能するようになりました。

詳細は、JDK-8240191 を参照してください。

3.1.3. 英語と他のロケールの間で、ローカライズされたタイムゾーン名が一致しません。

CLDR ロケールプロバイダーが提供する英語のタイムゾーン名は、COMPAT プロバイダーから置き換えられるのではなく、CLDR 仕様に従って正しく合成されるようになりました。

たとえば、このスタイル名は LONG スタイル名の省略形ではなく、GMT オフセット形式を生成します。

詳細は、JDK-8238914 を参照してください。

3.1.4. コンテナーの戻りコンテナー固有のデータ内の OperatingSystemMXBean メソッド

コンテナーまたは他の仮想化オペレーティングシステム環境でを実行すると、以下の **OperatingSystemMXBean** メソッドがコンテナー固有の情報 (利用可能な場合) を返します。それ以外の場合は、以下のホスト固有のデータを返します。

- getFreePhysicalMemorySize()

- getTotalPhysicalMemorySize()
- getFreeSwapSpaceSize()
- getTotalSwapSpaceSize()
- getSystemCpuLoad()

詳細は、JDK-8236876 を参照してください。

3.1.5. 信頼するルート認証局の追加: G4 証明書

Entrust ルート証明書が cacerts トラストストアに追加されました。

- エイリアス名: entrustrootcag4
  識別名: CN=Entrust Root Certification Authority - G4, OU="(c)2015 Entrust, Inc. - for authorized use only", OU=See www.entrust.net/legal-terms, O="Entrust, Inc.", C=US

詳細は、JDK-8250756 を参照してください。

3.1.6. 3 つの SSL をルート CA 証明書を追加

SSL 証明書の cacerts トラストストアに以下のルート証明書が追加されました。

- エイリアス名: sslrootrsaca
  識別名: CN=SSL.com Root Certification Authority RSA, O=SSL Corporation, L=Houston, ST=Texas, C=US
- エイリアス名: sslrootevrsaca
  識別名: CN=SSL.com EV Root Certification Authority RSA R2, O=SSL Corporation, L=Houston, ST=Texas, C=US
- エイリアス名: sslrooteccca
  識別名: CN=SSL.com Root Certification Authority ECC, O=SSL Corporation, L=Houston, ST=Texas, C=US

詳細は、JDK-8250860 を参照してください。

3.1.7. 弱いアルゴリズムが制限される前にユーザーに警告するように更新されました。

keytool および **jarsigner** ツールは、無効化される前に使用されている弱い暗号化アルゴリズムについてユーザーに警告するように更新されました。このツールは、SHA-1 ハッシュアルゴリズムと 1024 ビット RSA/DSA 鍵に関する警告を出力します。

詳細は、JDK-8244286 を参照してください。

3.1.8. TLS 署名スキームを設定するための新しいシステムプロパティー

OpenJDK の TLS 署名スキームをカスタマイズするために、新しいシステムプロパティーが 2 つ追加されました。**jdk.tls.client.SignatureSchemes**が TLS クライアントに追加され、**jdk.tls.server.SignatureSchemes** がサーバー側に追加されています。

各システムプロパティーには、TLS 接続に使用できる署名スキームを指定するサポートされる署名スキーム名のコンマ区切りリストが含まれます。

この名前は、**Java Security Standard Algorithm Names 仕様**の「署名スキーム」セクションで説明されています。

詳細は、JDK-8242147を参照してください。

3.1.9. krb5.conf での正規化のサポート

krb5.conf ファイルは、「canonicalize」フラグが JDK Kerberos 実装でサポートされるようになりました。**true** に設定すると、KDC サービス (AS プロトコル) への TGT 要求のクライアントから RFC 6806 名の正規化が要求されます。それ以外の場合は、デフォルトでは要求されません。

新しいデフォルト動作は以前のリリースとは異なり、TGT 要求で常に KDC サービスへの要求 (RFC 6806[1] のサポート) は **sun.security.krb5.disableReferrals** システムまたはセキュリティープロパティーで明示的に無効化されませんでした。

詳細は、JDK-8242059 を参照してください。

3.2. 非推奨の機能

3.2.1. TLS、CertPath、および Signed JAR の脆弱な名前付き曲線をデフォルトで無効に

名前付きの弱い曲線は、デフォルトで以下の **disabledAlgorithms** セキュリティープロパティーに追加することで無効にされます。

- jdk.tls.disabledAlgorithms
- jdk.certpath.disabledAlgorithms
- jdk.jar.disabledAlgorithms

Red Hat はアップストリームが提供する多くの曲線を常に削除しているので、本リリースで無効になっている曲線は以下のとおりです。

- secp256k1

以下の曲線は引き続き有効です。

- secp256r1
- secp384r1
- secp521r1
- X25519
- X448

名前付き曲線の弱い数を多数無効にする必要がある場合には、各 **disabledAlgorithms** プロパティーに個別の名前付き曲線を追加します。これを解決するために、新しいセキュリティープロパティー **jdk.disabled.namedCurves** が実装され、すべての **disabledAlgorithms** プロパティーに共通する名前付きの曲線を一覧表示できます。**disabledAlgorithms** プロパティーで新しいプロパティーを使用するには、フルプロパティー名の前に **include** キーワードを追加してください。この新しいプロパティーとは別の **disabledAlgorithms** プロパティーに、個別の名前付き曲線を追加できます。**disabledAlgorithms** プロパティーに追加できる他のプロパティーはありません。

名前付きの曲線を復元するには、すべての **include jdk.disabled.namedCurves** セキュリティープロパティーから、またはすべての **disabledAlgorithms** セキュリティープロパティーからを削除します。1 つ以上の曲線を復元するには、**jdk.disabled.namedCurves** プロパティーから特定の名前付きの曲線を削除します。

詳細は、JDK-8236730 を参照してください。

3.2.2. US/Pacific-New ゾーン名 tzdata2020b の一部として削除されました。

以下の JDK の tzdata2020b への更新で、long-obsolete ファイルが pacificnew および systemv から削除されています。その結果、pacificnew データファイルで宣言された「US/Pacific-New」ゾーン名が利用できなくなりました。

更新に関する情報は、https://mm.icann.org/pipermail/tz-announce/2020-October/000059.html で確認できます。

詳細は、JDK-8254177 を参照してください。

第4章 本リリースに関連するアドバイザリー

本リリースに含まれるバグ修正、および CVE の修正については、以下のアドバイザリーが発行されています。

- RHSA-2020:4316-05
- RHSA-2020:4305-05
- RHSA-2020:4306-05
- RHSA-2020:4307-02

Revised on 2020-11-02 08:56:49 CET