



Red Hat JBoss Enterprise Application Platform 7.3

7.3.0 リリースノート

Red Hat JBoss Enterprise Application Platform 7.3 向け

Red Hat JBoss Enterprise Application Platform 7.3 7.3.0 リリースノート

Red Hat JBoss Enterprise Application Platform 7.3 向け

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律上の通知

Copyright © 2021 | You need to change the HOLDER entity in the en-US/7.3.0_Release_Notes.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

本リリースノートには、Red Hat JBoss Enterprise Application Platform 7.3 に関する重要な情報が含まれています。

目次

第1章 JBOSS EAP 7.3 の認定ステータス	5
JBoss EAP 7.3 は Jakarta EE 8 と互換性のある実装です。	5
第2章 サポートされる構成	6
データベースおよびデータベースコネクタ	6
Java Development Kit	6
第3章 新機能および改良された機能	7
3.1. セキュリティー	7
Elytron 監査ロギングのパフォーマンスおよび信頼性の調整	7
JwtValidator の改良	7
デフォルトの SSLContext	7
Elytron を使用した JASPI (Java Authentication SPI for Containers) のセキュリティー	7
Server SSL Server Name Indication (SNI) コンテキスト	7
キーストアタイプの自動検出	7
Elytron での Java EE Security API サポート	8
Elytron でのサイレント BASIC 認証	8
Elytron でプロパティーベースのセキュリティーレルムをファイルシステムレルムに移行するユーティリティー	8
JDBC レルムでの 16 進数エンコーディングのサポート	8
JDBC レルムでの Modular Crypt Passwords のサポート	8
集約レルムでの承認に複数のセキュリティーレルムを組み合わせる機能	9
プリンシパルとして X.509 証明書からのサブジェクトの別名テキストを使用できる機能	9
集約エビデンスデコーダーと複数のエビデンスデコーダーを組み合わせられる機能	9
OCSP を使用した証明書失効機能	9
syslog 監査ロギングの改良	9
マスクしたパスワードの Elytron サポート	9
Let's Encrypt からのサーバー証明書の取得	9
集約レルムに追加されたプリンシパルトランスフォーマー	10
Elytron リソース filesystem-realm サポート	10
本リリースではデフォルトで有効になっている AJP コネクタ	10
HTTP 管理エンドポイントのカスタムヘッダー	10
3.2. サーバー管理	10
Maven アーティファクトおよびモジュール名の変更	10
Eclipse MicroProfile Metrics のサポート	10
ホストコントローラーによって管理されるサーバーの一時停止	10
Prometheus 形式の JBoss EAP サブシステムメトリクスのサポート	11
3.3. 管理 CLI	11
出力の一時停止を無効にします。	11
Liveness と Readiness プローブの MicroProfile Health Check 2.0.1 属性アップグレード	11
3.4. 管理コンソール	11
管理コンソールからのソケットログハンドラーの設定	11
管理コンソールからのロギングプロファイルログの表示	11
管理コンソールからのアクティブ管理操作の表示	11
ModCluster サブシステムで使用できる 2 つの新しいリソース	12
管理コンソールからの SSL SNI コンテキストの設定	12
現在進行中の操作以外の表示	12
ModCluster プロキシの設定	12
管理コンソールからのトラストマネージャーの再初期化	12
管理コンソールからの JASPI 認証の設定	12
管理コンソールからのリモート ActiveMQ サーバーリソースの設定	13
管理コンソールからのサーバーのソケットバインディング名とオープンポートの表示	13

管理コンソール対応のランタイム操作	13
Let's Encrypt アカウントの設定	13
管理コンソールを使用したキーストア認証局の設定	14
管理コンソールを使用した MicroProfile メトリクスの設定	14
SSL ウィザードでの Let's Encrypt CA から証明書の取得	14
管理コンソールタイトルのカスタマイズ機能	14
3.5. WEB SERVER	15
コンソールアクセスロギング	15
HttpServletRequest.getServletPath メソッドの動作の変更	15
3.6. LOGGING	15
Syslog メッセージのフォーマット機能	15
カスタムログフィルター	15
3.7. DEPLOYMENTS (デプロイメント)	15
デプロイメントに基づくモジュールの表示	15
3.8. EJB	15
メッセージ駆動 Bean の複数の配信グループのサポート	15
クライアントおよびサーバーインターセプター	15
3.9. クラスタリング	16
mod_cluster サブシステムの新しい属性の初期負荷	16
プライマリーシングルトンプロバイダーの特定機能	16
分散可能なセッションマネージャー呼び出しを指定する機能	16
新しいプライマリープロバイダーのシングルトンサービスプロバイダーを通知する機能	16
分散可能な Web セッション設定のための分散可能な Web サブシステム	16
リモート Red Hat Data Grid クラスタにセッションデータを保存する機能	16
ランク付けされた複数のルートセッションアフィニティを指定する機能	16
3.10. INFINISPAN	17
リモートキャッシュコンテナの統計の有効化	17
3.11. WEB サービス	17
オプションの <T> RESTEasy に利用できるパラメータータイプ	17
RESTEasy での HTTP プロキシのサポート	17
クライアントプロバイダーの処理の無効化	17
3.12. MESSAGING	17
resourceAdapter 要素を使用したリモート Artemis ベースのブローカーの JMS リソースの設定	18
メッセージングサーバーのグローバルリソースの設定	18
メッセージジャーナルファイルを開くためのタイムアウト値の設定	18
Artemis ロギングコードの変更	18
宛先名の接頭辞を省略します。	18
ロードバランサーのメッセージング機能強化	18
Apache Artemis に追加された処理済みメッセージ統計	18
スタンドアロン JMS クライアントでの JGroups との検出グループの使用	19
3.13. OPENSIFT	19
Galleon Feature-packs によるメモリーフットプリントの軽減	19
OpenShift でのアプリケーションディプロイメントを自動化する EAP オペレーター	19
安全なトランザクションリカバリーおよび EJB リモートリングの EAP オペレーター	19
JVM メモリー設定の計算	19
セキュアなアーティファクトリポジトリミラー URL	20
新バージョンの ASYM-ENCRYPT	20
IBM System Z での JBoss EAP のサポート	20
3.14. クイックスタートおよび BOM	20
JAX-WS の新しいクライアント BOM	20
Jakarta EE 8 の BOM への変更	20

管理コンソールを使用したヘルスチェックの確認	21
管理 CLI を使用したカスタムモジュールの追加および削除	21
MicroProfile Rest Client 1.3.2 以降のサポート	21
4.1. MICROPROFILE OPENTRACING 1.3.0 のサポート	21
4.2. MICROPROFILE METRICS 2.0.1 のサポート	21
第5章 サポートされない機能および非推奨の機能	22
5.1. サポートされない機能	22
データベースおよびデータベースコネクタ	22
OpenShift JDK 11 イメージの内部データソースおよびドライバー	22
Eclipse MicroProfile 機能	22
5.2. 非推奨の機能	22
5.2.1. プラットフォームおよび機能	23
オペレーティングシステムおよび関連する web サーバー	23
Java APIs	23
RHEL 6 AMI イメージ	23
OpenShift 自動化トランザクションリカバリ機能	23
第6章 解決した問題	25
第7章 修正された CVE	26
第8章 既知の問題	27

第1章 JBOSS EAP 7.3 の認定ステータス

JBoss EAP 7.3 は Jakarta EE 8 と互換性のある実装です。

JBoss EAP 7.3 リリースは、Web Profileと Full Platform 仕様の Jakarta EE 8 対応実装です。このリリースは、以前のリリースのような Java EE 8 Web Profile および Full Platform 仕様の認定実装です。

Jakarta EE の詳細は、「[Jakarta EE FAQ](#)」を参照してください。

第2章 サポートされる構成

以下の構成が JBoss EAP 7.3 で新たにサポートされます。

データベースおよびデータベースコネクタ

[Red Hat カスタマーポータル](#)の [Red Hat JBoss Enterprise Application Platform\(EAP\)7 でサポートされる構成](#) に関するナレッジベースの記事では、JBoss EAP 7.3 リリースの一環としてテストされたデータベースおよびデータベースコネクタを一覧表示しています。

Java Development Kit

- Eclipse OpenJ9 (JDK 11)

第3章 新機能および改良された機能

3.1. セキュリティー

Elytron 監査ロギングのパフォーマンスおよび信頼性の調整

JBoss EAP 7.2 では、Elytron ファイル監査ロギングの **synchronized** 属性により、出力ストリームをフラッシュし、ファイル記述子をすべての監査イベントの後に同期するかどうかが定義されていました。

本リリースでは、ストリームのフラッシュとファイル同期を分離するための新しい **autoflush** 属性が導入されました。これにより、Elytron 監査ロギングのパフォーマンスおよび信頼性を詳細に調整できるようになりました。

Elytron 監査ロギングの設定に関する詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Elytron Audit Logging](#)」を参照してください。

JwtValidator の改良

本リリースでは、JwtValidator に複数のキーとリモート公開鍵のサポートが追加されました。**key-store** 属性が、**public-key** の代替として使用できる **certificate** 属性と組み合わせ可能になりました。**client-ssl-context** 属性は、リモート [JSON Web Key\(JWK\)](#)に使用する [SSL コンテキストを定義します](#)。これにより、**jku** (JSON Key URL) ヘッダーパラメーターの URL を使用して、トークンの検証用に公開鍵を取得できます。

詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[token-realm jwt Attributes](#)」の表を参照してください。

デフォルトの SSLContext

今回のリリースでは、デフォルトの SSLContext を起動時に登録するようになりました。これは、デフォルトのコンテキストを利用可能なライブラリーで使用できます。

詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Default SSLContext](#)」を参照してください。

Elytron を使用した JASPI (Java Authentication SPI for Containers) のセキュリティー

このリリースの **elytron** サブシステムは、JASPI (Java Authentication SPI for Containers) からの **Servlet** プロファイル実装を行うことができます。これにより、Elytron のセキュリティー機能とのより密接な統合が可能になります。

詳細は、JBoss EAP 『[開発ガイド](#)』の「[Elytron を使用した Java Authentication SPI\(JASPI\)セキュリティーの設定](#)」を参照してください。

Server SSL Server Name Indication (SNI) コンテキスト

本リリースの **server-ssl-sni-context** は、サーバー側の SNI 一致に使用されます。与えられたホスト名がいずれも一致しない場合のデフォルトとともに、ホスト名を SSL コンテキストに関連付けるためのマッチングルールを利用できます。

詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Using a server-ssl-sni-context](#)」を参照してください。

キーストアタイプの自動検出

以下のキーストアタイプが自動的に検出されるようになりました。

- JKS
- JCEKS
- PKCS12

- BKS
- BCFKS
- UBER

その他のキーストタイプは手動で指定する必要があります。

詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Elytron Subsystem Components Reference](#)」を参照してください。

Elytron での Java EE Security API サポート

elytron サブシステムは JSR 375 で定義した Java EE Security API に対応するようになりました。

Java EE Security API は、認証およびアイデンティティストアの移植可能なプラグインインターフェースと、プログラムによるセキュリティのアクセスポイントを提供する新しい injectable-type **SecurityContext** インターフェースを定義します。これらの API のビルトイン実装を使用するか、カスタム実装を定義することができます。仕様の詳細は、「[Java EE Security API Specification](#)」を参照してください。

管理 CLI では、最小限の設定手順で、**elytron** サブシステムにおいて Java EE Security API を有効にできます。

Java EE Security API の有効化に関する詳細は、『[開発ガイド](#)』の「[Java EE Security API について](#)」を参照してください。

Elytron でのサイレント BASIC 認証

elytron サブシステムを設定することで、サイレント **BASIC** 認証を実行できるようになりました。

サイレント認証が有効になっていると、ユーザーの要求に承認ヘッダーが含まれていない場合でも、Web アプリケーションにアクセスする際にユーザーログインが求められません。

サイレント **BASIC** 認証の有効化に関する情報は、『[How to Configure Identity Management](#)』の「[Configure Web Applications to Use Elytron or Legacy Security for Authentication](#)」を参照してください。

Elytron でプロパティーベースのセキュリティレルムをファイルシステムレルムに移行するユーティリティ

elytron-tool.sh ツールの **filesystem-realm** コマンドを使用して、レガシープロパティーベースのセキュリティレルムを Elytron のファイルシステムベースのレルムに移行できるようになりました。

ファイルシステムベースのレルムは、ユーザー ID を保存するための、Elytron で使用されるファイルシステムベースのアイデンティティストアです。**filesystem-realm** コマンドは、**properties-realm** ファイルを **filesystem-realm** に移行します。また、このレルムとセキュリティドメインを **elytron** サブシステムに追加するコマンドも生成します。

filesystem-realm コマンドの詳細は、JBoss EAP 『[移行ガイド](#)』の「[Migrate to Filesystem-based Security Realm Using the filesystem-realm Command](#)」を参照してください。

JDBC レルムでの 16 進数エンコーディングのサポート

Elytron は、JDBC レルムのパスワードハッシュアルゴリズムに対する 16 進数エンコーディングに対応するようになりました。

詳細は、JBoss EAP 『[How to Configure Identity Management](#)』の「[Password Mappers](#)」を参照してください。

JDBC レルムでの Modular Crypt Passwords のサポート

Modular Crypt パスワードエンコーディングが JDBC レルムでサポートされるようになりました。

Modular Crypt エンコーディングでは、パスワードタイプ、ハッシュまたはダイジェスト、ソルト (salt)、反復カウント (iteration count) などの複数の情報が単一文字列でエンコードできるようになります。

詳細は、JBoss EAP 『[How to Configure Identity Management](#)』の「[Password Mappers](#)」を参照してください。

集約レルムでの承認に複数のセキュリティーレルムを組み合わせる機能
aggregate-realms 属性持つ **authorization-realms** での承認に複数のセキュリティーレルムを使用できるようになりました。

詳細は、JBoss EAP 『[How to Configure Identity Management](#)』ガイドの「[Configure Authentication and Authorization Using Multiple Identity Stores](#)」を参照してください。

プリンシパルとして X.509 証明書からのサブジェクトの別名テキストテンションを使用できる機能

証明書に関連するプリンシパルとして X.509 証明書からのサブジェクトの別名を使用するように、エビデンスデコーダーを設定できるようになりました。

詳細は、『[How to Configure Server Security](#)』ガイドの「[Configuring Evidence Decoder for X.509 Certificate with Subject Alternative Name Extension](#)」を参照してください。

集約エビデンスデコーダーと複数のエビデンスデコーダーを組み合わせられる機能
Elytron では、集約型デコーダーを利用して、2 つ以上のエビデンスデコーダーを単一のデコーダーに統合できます。

詳細は、『[How to Configure Server Security](#)』ガイドの「[Configuring an Aggregate Evidence Decoder](#)」を参照してください。

OCSP を使用した証明書失効機能

Elytron は、undertow がロードバランサーとして使用される場合に、[RFC6960](#) に定義される Online Certificate Status Protocol(OCSP)を使用して証明書失効機能を提供するようになりました。

詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Using Online Certificate Status Protocol for Certificate Revocation](#)」を参照してください。

syslog 監査ロギングの改良

Elytron syslog 監査ロギングが、監査イベントを記述するために [RFC5424](#) および [RFC 3164](#) で定義されたログ形式をサポートするようになりました。

新しい属性 **reconnect-attempts** では、UDP を使用する際に接続を閉じる前に Elytron が連続メッセージを syslog サーバーに送信する最大試行回数を設定できるようになりました。

機能拡張に関する詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Syslog Audit Logging](#)」を参照してください。

マスクしたパスワードの Elytron サポート

Elytron の元の実装は、マスクされたパスワードに対応していませんでした。

JBoss EAP 7.3 では、マスクされたパスワードタイプは PicketBox パスワードとの後方互換性として対応しています。

Let's Encrypt からのサーバー証明書の取得

JBoss EAP 7.2 では、SSL の管理を行うために基本的な CLI コマンドが追加されていました。

JBoss EAP 7.3 では、これらのコマンドは、Let's Encrypt 認証局からサーバー証明書を取得するように強化されています。

集約レルムに追加されたプリンシパルトランスフォーマー

Elytron には集約セキュリティレルムが含まれます。これは、認証レルムと1つ以上の承認レルムなど、2つ以上のレルムを組み合わせたものです。

JBoss EAP 7.2 以前では、認証 ID の読み込み後および承認 ID の読み込み前でプリンシパルを変換する機能はありませんでした。

現在では、集約レルムは、マッパーの設定で定義されたプリンシパルトランスフォーマーで設定でき、この変換を実行できます。

Elytron リソース `filesystem-realm` サポート

JBoss EAP 7.1 および 7.2 では、`filesystem-realm` はテクノロジープレビューとして利用できました。

本リリースでは、ファイルシステムでサポートされる Elytron リソース定義として `filesystem-realm` がサポートされるようになりました。

本リリースではデフォルトで有効になっている AJP コネクタ

JBoss EAP 7.3 では `undertow` サブシステムの AJP コネクタはデフォルトで有効になっています。AJP コネクタはこのサブシステムの潜在的なセキュリティリスクとして特定されました。

この [リスク](https://access.redhat.com/solutions/4851251) を解決するための推奨される方法は、<https://access.redhat.com/solutions/4851251> を参照してください。

HTTP 管理エンドポイントのカスタムヘッダー

以前のリリースの JBoss EAP には、管理インターフェースでエンドポイントのカスタム HTTP ヘッダーを定義する機能が含まれませんでした。

JBoss EAP 7.3 では、新しい属性 `constant-headers` が HTTP 管理インターフェースリソース定義に追加されます。管理者は、この属性を使用して JBoss EAP の追加 HTTP ヘッダーを指定し、HTTP 管理インターフェースに対するリクエストに応答したときに返すことができます。

3.2. サーバー管理

Maven アーティファクトおよびモジュール名の変更

JBoss EAP 7.3 アーティファクト `org.jboss.resteasy:resteasy-validator-provider-11` は、`org.jboss.resteasy:resteasy-validator-provider` として名前が変更されています。

また、モジュール `org.jboss.resteasy:resteasy-validator-provider-11` は現在、`org.jboss.resteasy:resteasy-validator` のエイリアスとして分類されるようになりました。新しく作成したアプリケーションは、モジュール `org.jboss.resteasy:resteasy-validator` を参照します。既存のアプリケーションは依然として、その機能に影響を及ぼさずにエイリアスを参照できます。

Eclipse MicroProfile Metrics のサポート

本リリースでは、`microprofile-metrics-smallrye` サブシステムを使用して `Eclipse MicroProfile Metrics` 機能を提供する `SmallRye Metrics コンポーネント` が含まれるようになりました。このサブシステムは、JBoss EAP インスタンスのモニタリングデータを提供するために使用されます。これはデフォルトで有効です。

詳細は、JBoss EAP 『設定ガイド』の「[Eclipse MicroProfile Metrics](#)」を参照してください。

ホストコントローラーによって管理されるサーバーの一時停止

本リリースでは、管理対象ドメインのホストレベルでサーバーをサスペンドおよび再開する機能を利用できます。

詳細は、JBoss EAP 『[設定ガイド](#)』の「[Suspend Servers](#)」を参照してください。

Prometheus 形式の JBoss EAP サブシステムメトリクスのサポート

Eclipse MicroProfile Metrics 機能は、JBoss EAP インスタンスのモニタリングデータを提供するために使用されます。本リリースでは、SmallRye Metrics コンポーネントが拡張され、Prometheus 形式で JBoss EAP メトリクスを提供します。

Eclipse MicroProfile Metrics の詳細は、JBoss EAP 『[設定ガイド](#)』の「[Eclipse MicroProfile Metrics](#)」を参照してください。

3.3. 管理 CLI

出力の一時停止を無効にします。

デフォルトでは、JBoss EAP 管理 CLI は出力ページの表示後に一時停止するため、コマンドの出力の閲覧および検索が可能です。現在では、この動作を無効にすることができます。また、**--no-output-paging** 引数で管理 CLI を起動するか、**EAP_HOME/bin/jboss-cli.xml** ファイルで **output-paging** 要素を **true** に設定することで、出力全体を即座に表示できるようになりました。

Liveness と Readiness プロブの MicroProfile Health Check 2.0.1 属性アップグレード

以前のリリースでは、管理 CLI を使用して **liveness** および **readiness** プロブの MicroProfile ヘルスステータスを個別に取得することはできませんでした。

現在では、**liveness** および **readiness** プロブのステータスは、**:check-live** および **:check-ready** 操作を使用するか、**/health/live** および **/health/ready** HTTP エンドポイントを使用することで個別に取得できるようになりました。

現在では、既存の **/health** HTTP エンドポイントおよび **:check** 管理操作を使用して、**liveness** および **readiness** プロブの両方のすべてのチェックプロブを取得できるようになりました。

現時点では、現在の MicroProfile Health Check 2.0.1 機能は以前のバージョンと完全に後方互換性がありません。

詳細は、『[設定ガイド](#)』の「[MicroProfile Health Check](#)」を参照してください。

3.4. 管理コンソール

管理コンソールからのソケットログハンドラーの設定

現在では、管理コンソールを使用して、**Configuration** → **Subsystems** → **Logging** → **Configuration** の順に選択し、**表示** をクリックして **Handler** → **Socket Handler** を選択すると、ソケットログハンドラーを設定することができます。

詳細は、『[設定ガイド](#)』の「[Configure a Socket Log Handler](#)」を参照してください。

管理コンソールからのロギングプロファイルログの表示

現在では、**Runtime** → **Monitor** → **Log Files** → **Log File** にアクセスし、ログを表示したいロギングプロファイルの隣の**表示**をクリックすることで、管理コンソールからロギングプロファイルのログファイルを表示することができます。

管理コンソールからのアクティブ管理操作の表示

管理コンソール内の中央で、すべてのホストおよびサーバーのアクティブな操作を表示できるようになりました。

スタンドアロンサーバーを実行している場合は、**Runtime** → **Server** → **Monitor** → **Management Operations** の順にクリックし、**表示**をクリックします。

管理対象ドメインでは、**Runtime** → **Browse By** → **Management Operations** に移動して表示をクリックします。

ModCluster サブシステムで使用できる 2 つの新しいリソース

modcluster サブシステムには、**load-provider=dynamic** および **load-provider=simple** という 2 つの新しいリソースが含まれるようになりました。**dynamic-load-provider=configuration** リソースは、**load-provider=dynamic** のエイリアスです。

管理コンソールで **Configuration** → **Configuration** → **Profile** → **full-ha** または **ha** → **Modcluster** → **Proxy** → **default (ajp)** の順に選択し、表示をクリックすることで、相互排他的なリソースを表示できるようになりました。

詳細は、JBoss EAP 『[設定ガイド](#)』の「[ModCluster サブシステムの属性](#)」を参照してください。

管理コンソールからの SSL SNI コンテキストの設定

管理コンソールから SSL SNI コンテキストを設定するには、**Configurations** → **Subsystems** → **Security (Elytron)** → **Other Settings** の順に選択し、表示をクリックします。**SSL** → **Server SSL SNI** コンテキスト コンテキストの順にクリックしてコンテキストを追加、編集、または削除します。

詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Configuring SSL SNI Context](#)」を参照してください。

現在進行中の操作以外の表示

管理コンソールには、現在進行していない操作が発生したときに通知が表示されるようになりました。この通知は **Runtime** タブからアクセスできます。

スタンドアロンサーバーを実行している場合は、**Runtime** → **Monitor** → **Management Operations** に移動して表示をクリックします。**Cancel Non Progressing Operations** ボタンは、**Reload** ボタンの横にあるウィンドウの右上隅にあります。通知では、現在進行中の操作が一覧表示されます。

管理対象ドメインでは、**Runtime** タブを選択するところで、この通知にアクセスできます。[参照](#)の列で、**管理操作**をクリックします。

ModCluster プロキシの設定

本リリースでは、コンソールの **Configuration** タブで使用できる **modcluster** サブスキームのマルチサーバーサポートが導入されました。Modcluster 列のタイトルは **Proxy** で、ここでは **/subsystem=modcluster/proxy=*** 下のプロキシが一覧表示されています。

追加 および **削除** 操作では、プロキシの管理が簡素化されます。また、表示操作では、選択したプロキシの設定オプションが開きます。

詳細は、JBoss EAP 『[設定ガイド](#)』の「[ModCluster システム属性](#)」を参照してください。

管理コンソールからのトラストマネージャーの再初期化

管理コンソールから JBoss EAP で設定したトラストマネージャーを再初期化するには、**Runtime** → **Monitor** → **Security (Elytron)** → **SSL** の順に選択し、表示をクリックして **Trust Manager** を選択します。詳細は、JBoss EAP 『[How to Configure Server Security](#)』の「[Reinitializing a Trust Manager from the management console](#)」を参照してください。

管理コンソールからの JASPI 認証の設定

JASPI 認証モジュールの設定は、**Configuration** → **Subsystem** → **Security (Elytron)** → **Other Settings** の順に選択し、表示をクリックすることで管理コンソールから行えるようになりました。**Other Settings** → **JASPI Configuration** の順にクリックしてモジュールを設定します。

詳細は、JBoss EAP 『[セキュリティアーキテクチャー](#)』の「[セキュリティ 管理](#)」を参照してください。

管理コンソールからのリモート **ActiveMQ** サーバーリソースの設定

管理コンソールから以下のリモート ActiveMQ サーバーリソースを設定することができるようになりました。

- 汎用コネクタ
- VM コネクタ
- HTTP コネクタ
- リモートコネクタ
- 検出グループ
- 接続ファクトリー
- プールされた接続ファクトリー
- 外部 JMS キュー
- 外部 JMS トピック

詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[Configure Remote ActiveMQ Server Resources Using the Management Console](#)」を参照してください。

管理コンソールからのサーバーのソケットバインディング名とオープンポートの表示

ソケットバインディング名と、サーバーのオープンポートが管理コンソールから確認できるようになりました。この情報は、サーバーの状態が以下のときに表示されます。

- **running**
- **reload-required**
- **restart-required**

詳細は、JBoss EAP 『[設定ガイド](#)』の「[Viewing Socket Bindings and Open Ports for a Server](#)」を参照してください。

管理コンソール対応のランタイム操作

管理 CLI のみを使用して実行できるランタイム操作の一部が管理コンソールでも利用できるようになりました。

詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[Runtime Operations Using the Management Console](#)」を参照してください。

Let's Encrypt アカウントの設定

管理コンソールを使用して Let's Encrypt アカウントを設定することができるようになりました。以下の設定が可能です。

- 認証局でアカウントを作成します。
- 認証局アカウントを非アクティブにします。
- アカウントを更新します。
- 認証局のアカウント情報を表示します。

- 認証局アカウントキーを変更します。

Let's Encrypt アカウントの設定に関する詳細は、JBoss EAP『[How to Configure Server Security](#)』の「[Let's Encrypt アカウントの設定](#)」を参照してください。

管理コンソールを使用したキーストア認証局の設定

以下のキーストアの認証局設定が、管理コンソールで実行できるようになりました。

- エントリーのエイリアスを変更します。
- キーストアのエントリーからファイルに証明書をエクスポートします。
- 証明書の署名要求を生成します。
- キーストアからエイリアスを削除します。
- エイリアスに関連付けられた証明書の詳細を表示します。
- エイリアスに関連付けられた証明書を取り消します。
- 証明書の更新が必要かどうかを判断します。

キーストア証明書管理の詳細は、JBoss EAP『[How to Configure Server Security](#)』の「[Keystore Certificate Authority Operations Using the Management Console](#)」を参照してください。

管理コンソールを使用した MicroProfile メトリクスの設定

管理コンソールで、MicroProfile メトリクスを設定できるようになりました。

管理コンソールで利用可能な設定は次のとおりです。

- メトリクスの公開の有効化または無効化
- 接頭辞の編集
- セキュリティーの有効化または無効化
- 必須でないフィールドの、初期またはデフォルト値へのリセット

MicroProfile メトリクスの設定に関する詳細は、JBoss EAP『[設定ガイド](#)』の「[管理コンソールを使用した MicroProfile メトリクスの設定](#)」を参照してください。

SSL ウィザードでの Let's Encrypt CA から証明書の取得

SSL ウィザードで、Let's Encrypt Certificate Authority から証明書を取得できるようになりました。

詳細は、以下のリンクを参照してください。

- JBoss EAP『[How to Configure Server Security](#)』の「[Enable SSL Using the Management Console for applications](#)」を参照してください。
- JBoss EAP『[How to Configure Server Security](#)』の「[Enable SSL Using the Management Console for management interface](#)」を参照してください。

管理コンソールタイトルのカスタマイズ機能

JBoss EAP インスタンスのそれぞれを一目で特定できるように、管理コンソールのタイトルをカスタマイズできるようになりました。

詳細は、JBoss EAP 『設定ガイド』の「[Customizing the Management Console Title](#)」を参照してください。

3.5. WEB SERVER

コンソールアクセスロギング

コンソールにログデータを出力する新しい機能が追加されました。コンソールアクセスロギングデータは、1行の SON 構造データとして **stdout** に書き込まれます。

詳細は、JBoss EAP 『設定ガイド』の「[サーバーの設定](#)」を参照してください。

HttpServletRequest.getServletPath メソッドの動作の変更

HttpServletRequest.getServletPath メソッドは、JBoss Web サブシステムでは Undertow とは異なる動作をします。特に Undertow では、このメソッドはアクション名ではなく JSP 名を返します。

JBoss Web サブシステムで行ったように HttpServletRequest.getServletPath メソッドを設定する属性が追加されました。

3.6. LOGGING

Syslog メッセージのフォーマット機能

管理コンソールで **named-formatter** 属性を設定できるようになりました。

カスタムログフィルター

JBoss EAP 7.2 以前では、ユーザーが使用できるフィルターは、与えられたログフィルターに制限されていました。

JBoss EAP 7.3 では、ユーザーはカスタムログフィルターを実装できます。

3.7. DEPLOYMENTS (デプロイメント)

デプロイメントに基づくモジュールの表示

list-modules 管理操作で、デプロイメントに基づいてモジュールのリストを表示できるようになりました。

list-modules 管理操作の使用に関する詳細は、『[開発ガイド](#)』の「[Display Modules by Deployment](#)」セクションを参照してください。

3.8. EJB

メッセージ駆動 Bean の複数の配信グループのサポート

メッセージ駆動 Bean (MDB) は、複数の配信グループに属せるようになりました。メッセージ配信は、MDB が属するすべての配信グループがアクティブの場合にのみ有効になります。

詳細は、JBoss EAP 『[Developing EJB Applications](#)』ガイドの「[Delivery Groups](#)」を参照してください。

クライアントおよびサーバーインターセプター

JBoss EAP 7.2 以前は、コンテナで設定した EJB インターセプターのみに対応していました。

JBoss EAP 7.3 では、クライアントインターセプターおよびサーバーインターセプターに対応するようになりました。クライアントおよびサーバーのインターセプターは、グローバルに設定できます。

詳細は、JBoss EAP 『[Developing EJB Applications](#)』ガイドの「[Custom Interceptors](#)」を参照してください。

3.9. クラスタリング

mod_cluster サブシステムの新しい属性の初期負荷

mod_cluster サブシステムが新しい属性 **initial-load** を定義するようになりました。

initial-load 属性は、新しく加わるノードの負荷値を段階的に増大させ、クラスターに加わる際のオーバーロードを回避するのに役立ちます。

この属性に関する詳細は、JBoss EAP 『[設定ガイド](#)』の「[ModCluster サブシステムの属性](#)」を参照してください。

プライマリーシングルトンプロバイダーの特定機能

プライマリーシングルトンプロバイダーは、各シングルトンデプロイメントまたは特定のシングルトンポリシーから作成されたサービスに対して、**singleton** サブシステムが公開するランタイムリソースを使って判断できるようになりました。

詳細は、JBoss EAP 『[開発ガイド](#)』の「[Determine the Primary Singleton Service Provider Using the CLI](#)」を参照してください。

分散可能なセッションマネージャー呼び出しを指定する機能

META-INF/jboss-all.xml 設定ファイルで **<shared-session-config>** の下に **<distributable/>** タグを追加することで、サブデプロイメント間でのセッション共有を行う際に使用する分散可能なセッションマネージャーを指定できるようになりました。

詳細は、JBoss EAP 『[開発ガイド](#)』の「[エンタープライズアーカイブのサブデプロイメント間で共有するセッションの設定](#)」を参照してください。

新しいプライマリープロバイダーのシングルトンサービスプロバイダーを通知する機能

登録された **SingletonElectionListener** を持つクラスターの各メンバーは、新しいプライマリーシングルトンサービスプロバイダーが選ばれると通知を受信します。

詳細は、JBoss EAP 『[開発ガイド](#)』の「[HA Singleton Service Election Listener](#)」を参照してください。

分散可能な Web セッション設定のための分散可能な Web サブシステム

JBoss EAP の新しい **distributable-web** サブシステムにより、柔軟で分散可能な Web セッション設定が容易になります。サブシステムは **jboss-web.xml** の **<replication-config>** 要素の使用を終了します。

詳細は、JBoss EAP 『[開発ガイド](#)』の「[分散可能な Web セッション設定の distributable-web サブシステム](#)」および『[移行ガイド](#)』の「[デフォルトの分散可能なセッション管理動作のオーバーライド](#)」を参照してください。

リモート Red Hat Data Grid クラスターにセッションデータを保存する機能

distributable-web サブシステムは、HotRod プロトコルを使用して、リモート Red Hat Data Grid クラスターに Web セッションデータを保存するように設定できるようになりました。リモートクラスターに Web セッションデータを格納すると、キャッシュレイヤーがアプリケーションサーバーとは独立して拡張できるようになります。

distributable-web サブシステムの設定に関する詳細は、JBoss EAP 『[開発ガイド](#)』の「[Storing Web Session Data In a Remote Red Hat Data Grid](#)」を参照してください。

ランク付けされた複数のルートセッションアフィニティを指定する機能

ノードの順序付けされたリストとしてセッションアフィニティを指定できるようになりました。ロー

ドバランサーが複数の、順序付けされたルートに対応していれば、プライマリーノードに障害が発生した場合に、定義した順序で最適なノードを選択できます。また、特定のセッションの主な所有者が非アクティブであれば、無限なフェイルオーバーターゲット確立されます。

ランク付けされたアフィニティーオプションの詳細は、JBoss EAP 『開発ガイド』の「分散可能な Web セッション設定の `distributable-web` サブシステム」を参照してください。ロードバランサーでランク付けされたセッションアフィニティーを有効にする方法は、JBoss EAP 『設定ガイド』の「[Enabling Ranked Session Affinity in Your Load Balancer](#)」を参照してください。

3.10. INFINISPAN

リモートキャッシュコンテナの統計の有効化

HotRod クライアントを使用してリモートキャッシュコンテナの統計を公開できるようになりました。`statistics-enabled` 属性は、管理 CLI を使用してリモートキャッシュコンテナに対して設定できます。詳細は、JBoss EAP 『設定ガイド』の「[リモートキャッシュコンテナの設定](#)」を参照してください。

3.11. WEB サービス

オプションの `<T>RESTEasy` に利用できるパラメータータイプ

RESTEasy は、ラッパーオブジェクトタイプとして以下の `java.util.Optional` パラメーターに対応するようになりました。

- `@QueryParam`
- `@MatrixParam`
- `@FormParam`
- `@HeaderParam`
- `@CookieParam`

詳細は、JBoss EAP 『[Developing Web Services Applications](#)』の「[java.util.Optional Parameter Types](#)」を参照してください。

RESTEasy での HTTP プロキシのサポート

RESTEasy の元の実装には HTTP プロキシのサポートが含まれていませんでした。

JBoss EAP 7.3 では、JAX-RS API (RESTful サービスの Java API) を使用して HTTP プロキシをクライアントビルダーで設定できます。

クライアントプロバイダーの処理の無効化

`@Provider` でアノテーションの付いたクライアントプロバイダーは、JAX-RSコンテナランタイムの各インスタンスに登録される必要があります。システムプロパティー

`resteasy.client.providers.annotations.disabled` は、`@Provider` アノテーションが付けられたクライアントプロバイダーのデフォルトの処理を無効にし、不要または重複したクライアントプロバイダー登録の問題を回避します。

詳細は、JBoss EAP 『[Developing Web Services Applications](#)』の「[Content Marshalling with @Provider Classes](#)」を参照してください。

3.12. MESSAGING

resourceAdapter 要素を使用したリモート Artemis ベースのブローカーの JMS リソースの設定

Red Hat AMQ 7 などのリモート Artemis ベースのブローカーの JMS リソースを設定するには、**@JMSConnectionFactoryDefinition** アノテーションまたは **@JMSDestinationDefinition** アノテーションを使用します。**resourceAdapter** 要素は、JMS リソースの作成に使用されるリソースアダプターを定義します。

詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[JMS Resources Configuration for a Remote Artemis-based Broker](#)」を参照してください。

メッセージングサーバーのグローバルリソースの設定

address-setting 要素の新しい属性は、メッセージングサーバーのグローバルリソース使用を制御するのに役に立ちます。詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[Configure Global Resource Usage for Messaging Servers](#)」を参照してください。

メッセージジャーナルファイルを開くためのタイムアウト値の設定

journal-file-open-timeout 属性を使用して、メッセージジャーナルファイルを開くタイムアウトを設定できるようになりました。

journal-file-open-timeout 属性の設定に関する詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[Configuring Message Journal Attributes](#)」を参照してください。

Artemis ログインコードの変更

Artemis コアプロトコルの Artemis ログインコードが変更になりましたが、Advanced Message Queuing Protocol (AMQP) コードは変わりません。これにより、これらのコードに基づいて問題の監視を行う際に問題が発生します。

このコードは AMQP と Artemis コアプロトコル間で複製されていたため、ログインコードが変更されました。

宛先名の接頭辞を省略します。

接続ファクトリーまたはプールされた接続ファクトリーを設定して、リモート Artemis サーバーと通信する際に、宛先名の接頭辞を省略することができます。このオプションは、互換モードではないリモート Artemis 2.x との通信を設定するときに使用します。

詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[Using the Integrated Artemis Resource Adapter for Remote Connections](#)」ステップ 3、「[Configuring the Artemis Resource Adapter to Connect to Red Hat AMQ](#)」ステップ 4 を参照してください。

ロードバランサーのメッセージング機能強化

静的 HTTP ロードバランサーの既存のサポートに加え、`mod_cluster` を使用するロードバランサーにも対応しました。詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[ロードバランサーの背後にあるメッセージング](#)」を参照してください。

ロードバランサーの背後にあるクラスターへのメッセージングに完全に対応するようになりました。HTTP ロードバランサーの背後でクラスターと通信するクライアントは、クラスタートポロジーではなく、最初の接続を再利用する必要があります。詳細は、JBoss EAP 『[Configuring Messaging](#)』の「[ロードバランサーの背後にあるメッセージング用のクライアント設定](#)」を参照してください。

Apache Artemis に追加された処理済みメッセージ統計

Apache Artemis プロジェクトが以下の統計を追加しました。

- 処理されたメッセージ
- 中断またはロールバックされたメッセージ

これらの統計は、以下の CLI コマンドを実行して JBoss EAP で使用できるようになりました。

```
/subsystem=messaging-activemq/jms-bridge=bridge:read-attribute(name=message-count)
```

```
/subsystem=messaging-activemq/jms-bridge=bridge:read-attribute(name=aborted-message-count)
```

スタンドアロン JMS クライアントでの **JGroups** との検出グループの使用
 スタンドアロン JMS クライアントでの JGroups との検出グループの使用は非推奨になりました。

検出グループは Netty UDP とのみ使用する必要があります。

3.13. OPENSIFT

Galleon Feature-packs によるメモリーフットプリントの軽減

メインの JBoss EAP for OpenShift のイメージ設定をカスタマイズすることで、必要な機能のみを含めることができます。このため、メモリーフットプリントを低減することができます。プロビジョニングツール **Galleon** は、JBoss EAP サーバーにある機能を制御するために選択できる複数のレイヤーを提供します。

さらに、JBoss EAP サーバーの Galleon プロビジョニングの際にインストールされる Galleon feature-packs としてカスタマイズした機能をパッケージ化できます。JBoss EAP for OpenShift Source-to-Image (S2I) の場合は、feature-packs をオフラインでビルドし、それらを Maven にデプロイし、**provisioning.xml** ファイルで参照することができます。

OpenShift でのアプリケーションディプロイメントを自動化する EAP オペレーター

JBoss EAP では、JBoss EAP 固有のコントローラーである EAP オペレーターを利用して、共通のデプロイメント関連のタスクを自動化できるようになりました。EAP オペレーターは、OpenShift クラスター管理者がオペレーターの検出、インストール、アップグレードに使用する GUI の OperatorHub を使用してインストールできます。

EAP オペレーターのインストール、アンインストール、および使用方法は、『[JBoss EAP for OpenShift Container Platform のスタートガイド](#)』の「[OpenShift でのアプリケーションディプロイメントを自動化する EAP Operator](#)」を参照してください。

安全なトランザクションリカバリーおよび EJB リモータリングの EAP オペレーター

EAP オペレーターは、レプリカを縮小し、削除するために **clean** として Pod をマークする前に、すべてのトランザクションが完了していることを検証することで、アプリケーションクラスターでの安全なトランザクションリカバリーを行えるようにします。

EAP オペレーターは、EJB リモータリングおよびトランザクションリカバリープロセッシングの適切な処理に **StatefulSet** を使用します。**StatefulSet** は Pod の再起動後もストレージおよびネットワークのホスト名を永続的に保持します。

EAP オペレーターを使用した安全なトランザクションの回復の詳細は、「[EAP Operator for Safe Transaction Recovery](#)」を参照してください。EAP オペレーターを使用した EJB リモータリングの設定に関する詳細は、『[JBoss EAP for OpenShift Container Platform のスタートガイド](#)』の「[OpenShift での EJB の設定](#)」を参照してください。

JVM メモリー設定の計算

JBoss EAP 7.2 以前は、JVM メモリー設定がコンテナ制限よりも高い状態には対応していませんでした。

OpenShift EAP イメージは、コンテナ制限が定義され、JVM メモリー設定がコンテナ制限よりも高い場合に、デフォルトの JVM メモリー設定を算出できるようになりました。

詳細は、『[JBoss EAP for OpenShift Container Platform のスタートガイド](#)』の「[JVM Memory Configuration](#)」を参照してください。

セキュアなアーティファクトリポジトリミラー URL

Maven リポジトリを使った "man-in-the-middle" 攻撃を回避するために、JBoss EAP ではアーティファクトリポジトリのミラー URL にセキュアな URL を使用する必要があります。

詳細は、『JBoss EAP for OpenShift Container Platform のスタートガイド』の「[アーティファクトリポジトリミラー](#)」を参照してください。

新バージョンの ASYM-ENCRYPT

JBoss EAP 7.3 には **ASYM_ENCRYPT** プロトコルの新しいバージョンが含まれています。以前のバージョンのプロトコルは非推奨になりました。**JGROUPS_CLUSTER_PASSWORD** 環境変数を指定する場合は、プロトコルの非推奨バージョンが使用され、警告が Pod ログに出力されます。

詳細は、JBoss EAP 『JBoss EAP for OpenShift Container Platform のスタートガイド』の「[Configuring ASYM_ENCRYPT](#)」を参照してください。

IBM System Z での JBoss EAP のサポート

JBoss EAP は、IBM System Z インフラストラクチャーでプロビジョニングされる OpenShift 環境でのみ s390x プラットフォームでサポートされます。システム Z での Red Hat Enterprise Linux のスタンドアロンインストールでの JBoss EAP の実行はサポートされていません。

3.14. クイックスタートおよび BOM

JAX-WS の新しいクライアント BOM

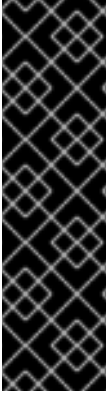
JBoss EAP では JAX-WS 依存関係を管理に新しいクライアント BOM (**wildfly-jaxws-client-bom**) を利用できるようになりました。プロジェクトに依存関係 **wildfly-jaxws-client-bom** を追加する方法は、JBoss EAP 『[開発ガイド](#)』の「[JBoss EAP クライアント BOM](#)」を参照してください。

Jakarta EE 8 の BOM への変更

Group ID **org.jboss.bom** の一部の JBoss EAP BOM は、JBoss EAP 7.3 の Jakarta EE 8 プラットフォームへの移行により置き換えられました。置き換えられた BOM をアプリケーションが使用する場合は、アプリケーションを JBoss EAP 7.3 リリースに移行するときにプロジェクト POM を更新し、新しい BOM のアーティファクト ID が含まれるようにします。

EAP BOM への変更に関する詳細は、JBoss EAP 『[移行ガイド](#)』の「[Jakarta EE 8 の BOM への変更](#)」を参照してください。

第4章 テクノロジーレビュー



重要

以下の設定および機能は、テクノロジーレビューとしてのみ提供されます。テクノロジーレビューの機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、Red Hat は本番環境での使用は推奨しません。テクノロジーレビューの機能は、最新の技術をいち早く提供して、開発段階で機能のテストやフィードバックの収集を可能にするために提供されません。

[テクノロジーレビュー機能のサポート範囲](#) については、Red Hat カスタマーポータル の「[テクノロジーレビュー機能のサポート範囲](#)」を参照してください。

管理コンソールを使用したヘルスチェックの確認

管理コンソールを使用してサーバーのヘルスチェックを検証できるようになりました。この機能は、JBoss EAP をスタンドアロンサーバーとして実行している場合のみ利用できます。

この機能の使用に関する詳細は、『[設定ガイド](#)』の「[Examine the Health Check Using the Management Console](#)」を参照してください。

管理 CLI を使用したカスタムモジュールの追加および削除

スタンドアロンサーバーの実行時に、**module** コマンドを使用してカスタムモジュールを追加および削除できます。このコマンドは、管理対象ドメイン環境での使用や、リモートで管理 CLI に接続する場合に適していません。

カスタムモジュールの作成に関する詳細は、JBoss EAP 『[設定ガイド](#)』の「[管理 CLI を使用したカスタムモジュールの作成](#)」を参照してください。

MicroProfile Rest Client 1.3.2 以降のサポート

JBoss EAP は MicroProfile Rest Client 1.3.2 以降に対応するようになりました。[MicroProfile Rest Client の詳細は、JBoss EAP 『\[Developing Web Services Applications\]\(#\)』の「\[MicroProfile REST Client\]\(#\)」を参照してください。](#)

以前のバージョンの MicroProfile REST クライアントを使用していた場合は、コードに更新をいくつか行う必要があります。コードの変更に関する詳細は、「[MicroProfile Rest Client Code で必要な変更](#)」を参照してください。

4.1. MICROPROFILE OPENTRACING 1.3.0 のサポート

JBoss EAP は MicroProfile OpenTracing 1.3.0 に対応するようになりました。**microprofile-opentracing-smallrye サブシステムの詳細は、『[設定ガイド](#)』の「[MicroProfile OpenTracing SmallRye サブシステムでのリクエストのトレース](#)」を参照してください。**

4.2. MICROPROFILE METRICS 2.0.1 のサポート

JBoss EAP は MicroProfile Metrics 2.0.1 に対応するようになりました。**microprofile-metrics-smallrye サブシステムに関する詳細は、『[設定ガイド](#)』の「[Eclipse MicroProfile Metrics](#)」を参照してください。**

第5章 サポートされない機能および非推奨の機能

5.1. サポートされない機能

メンテナンスコストの高さ、コミュニティでの関心の低さ、および高性能な代替ソリューションの存在などが理由で、一部の技術がサポート対象外になりました。

ここで設定したコンテンツにより、Red Hat で拘束力のある契約が発生したり、法的義務が強えられることはありません。この情報は、説明のみを目的としており、変更される可能性があります。

データベースおよびデータベースコネクタ

[Red Hat カスタマーポータル](#)の [Red Hat JBoss Enterprise Application Platform\(EAP\)7 でサポートされる構成](#) に関するナレッジベースの記事では、JBoss EAP 7.3 リリースの一環としてテストされたデータベースおよびデータベースコネクタを一覧表示しています。

JBoss EAP 7.3 リリースの一部としてテストされていないデータベースおよびデータベースコネクタを使用すると、JBoss EAP 7.3 インスタンスで問題が発生する可能性があります。

OpenShift JDK 11 イメージの内部データソースおよびドライバー

[Red Hat カスタマーポータル](#)の [Red Hat JBoss Enterprise Application Platform\(EAP\)7 でサポートされる構成](#) に関するナレッジベースの記事では、JBoss EAP 7.3 リリースの一環としてテストされた OpenShift JDK 11 イメージのデータベースおよびデータベースコネクタを一覧表示しています。

JBoss EAP アプリケーションのパフォーマンスを最適化するには、データベースベンダーから取得した JDBC ドライバーを使用します。

関連情報

- ドライバーのインストールに関する詳細は、『[JBoss EAP for OpenShift Container Platform のスタートガイド](#)』の「[モジュール、ドライバー、および汎用デプロイメント](#)」を参照してください。
- JBoss EAP で JDBC ドライバーを設定するための詳細は、JBoss EAP『[設定ガイド](#)』の「[JDBC ドライバー](#)」を参照してください。

Eclipse MicroProfile 機能

JBoss EAP 7.3 CD リリースおよび JBoss EAP 7.3 ベータリリースに含まれていた以下の Eclipse MicroProfile 機能は利用できなくなりました。

- microprofile-smallrye-health
- microprofile-smallrye-metrics
- microprofile-smallrye-config
- microprofile-smallrye-opentracing

JBoss EAP では、Eclipse MicroProfile API のサポートを、拡張メカニズムを使用して今後のリリースで提供しています。

5.2. 非推奨の機能

本リリースでは一部の機能が非推奨になりました。非推奨の機能には改良が加えられず、将来的 (通常は次のメジャーリリース) に削除される可能性があります。

Red Hat は標準のサポート条件に基づき、継続して完全サポートおよびバグ修正を提供します。Red Hat サポートポリシーの詳細は、Red Hat カスタマーポータルにある [Red Hat JBoss Middleware の製品アップデートとサポートポリシー](#) を参照してください。

非推奨となった機能の詳細は、Red Hat カスタマーポータルに [記載されている JBoss Enterprise Application Platform コンポーネント](#) の詳細を参照してください。

5.2.1. プラットフォームおよび機能

本リリースでは、以下のプラットフォームおよび機能のサポートが非推奨となりました。

オペレーティングシステムおよび関連する web サーバー

- Windows Server 2012 R2 および関連する IIS web サーバー
- JBoss EAP を管理するための Red Hat JBoss Operations Network (JON) の使用は非推奨となりました。
JBoss EAP を管理するための Red Hat JBoss Operations Network (JON) の使用は、JBoss EAP 7.2 リリースで非推奨となりました。JBoss EAP 7.3 リリースでは、依然として非推奨です。

JON ライフサイクルの移行サポートフェーズで利用できるサポートは制限されています。これに伴い、最新の JON プラグインの更新では、JBoss EAP 7.3 のテストとバグ修正は限られています。本番環境の JBoss EAP およびその他の Red Hat Middleware ワークロードの重要な JON 修正はライフサイクルのこのフェーズで提供されます。ただし、新しい JBoss EAP 機能に対するプラグインは更新されません。新しい機能は、プラグインで利用できないことがあります。また、互換性がなくなる可能性があります。

Jon 3.3 では、ユーザーは JON EAP Management プラグインパックのアップデート 11 以上をインストールし、JBoss EAP 7.3 と互換性を持たせる必要があります。

Java APIs

- SAAJ 1.3 は非推奨になり、JBoss EAP 7.3 では SAAJ 1.4 がデフォルトになります。
デフォルトの SAAJ バージョンの変更により、以前のリリースから JBoss EAP 7.3 への移行時にユーザー定義の SOAP ハンドラーで問題が発生する可能性があります。SAAJ バージョンが原因で発生する問題を解決するには、システムプロパティ `-Djboss.saaj.api.version=1.3` を、非推奨 API を使用するように設定します。

RHEL 6 AMI イメージ

Red Hat Enterprise Linux バージョン 6 AMI イメージは非推奨になりました。RHEL 6 AMI イメージは依然として含まれ、サポート対象ですが、非推奨の機能には改良が加えられず、将来的 (通常は次のメジャーリリース) に削除される可能性があります。Red Hat は標準のサポート条件に基づき、継続して完全サポートおよびバグ修正を提供します。



注記

現在、Red Hat Enterprise Linux バージョン 6 は、2020 年 11 月 30 日の時点で、メンテナンスサポートまたはメンテナンスサポート 2 (製品の終了日) に到達する予定です。

OpenShift 自動化トランザクションリカバリー機能

テクノロジープレビューとして提供される自動化トランザクションリカバリー機能が非推奨になりました。

サポートされる代替方法は、EAP オペレーターを使用してトランザクションから安全にリカバリーすることです。

以下の自動トランザクションリカバリーテクノロジープレビューアーティファクトも非推奨となりました。

- トランザクションリカバリーアプリケーションテンプレート
 - **eap73-tx-recovery-s2i**
 - **eap73-openjdk11-tx-recovery-s2i**
- OpenShift 用の **jta-crash-rec-eap7** クイックスタート

その他のリソース

- [安全なトランザクションリカバリーの EAP Operator](#)
- [OpenShift Quickstart Repository](#)

第6章 解決した問題

本リリースで解決した問題のリストは、『[Resolved Issues for JBoss EAP 7.3](#)』を参照してください。

第7章 修正された CVE

- [CVE-2018-7489](#): **jackson-databind**: CVE-2017-7525 の修正が不完全なため、c3p0 ライブラリーを介した安全でないシリアル化が許可されます
- [CVE-2018-1000632](#): **dom4j**: クラスでの XML インジェクションメソッド: addElement、addAttribute。これらは、XML ドキュメントの整合性に影響を与える可能性があります。
- [CVE-2019-9511](#): **undertow**: HTTP/2: データリクエストが多くなり、サービス拒否(DoS)が発生する
- [CVE-2019-9512](#): **undertow**: HTTP/2: PING フレームを使用したフラッドにより、無限のメモリー増大が発生
- [CVE-2019-9514](#): **undertow**: HTTP/2: HEADERS フレームを使用したフラッドにより、無限のメモリー増大が発生
- [CVE-2019-9515](#): **undertow**: HTTP/2: SETTINGS フレームを使用したフラッドにより、無限のメモリー増大が発生
- [CVE-2019-10219](#): **hibernate-validator**: safeHTML バリデーターが XSS を許可
- [CVE-2019-19343](#): **undertow**: リモート接続を無限に保持することによる Undertow HttpOpenListener のメモリーリーク
- [CVE-2019-14838](#): **wildfly-core**: デフォルトで 'Monitor'、'Auditor'、および 'Deployer' ユーザーの正しくない特権
- [CVE-2019-14885](#): **JBoss EAP**: vault システムプロパティのセキュリティー属性は、CLI 'reload' コマンドにおいて実行されます。
- [CVE-2019-16869](#): **netty**: HTTP ヘッダーのコロンの前に空白を誤って処理した HTTP リクエストスマグリング
- [CVE-2019-16942](#): **jackson-databind**: commons-dbcp パッケージのクラスにおけるシリアル化ガジェット
- [CVE-2019-16943](#): **jackson-databind**: commons-dbcp パッケージのクラスにおけるシリアル化ガジェット

第8章 既知の問題

本リリースでの [既知の問題の一覧](#)は、『[Known Issues for JBoss EAP 7.3](#)』を参照してください。

この他に、以下に注意してください。

- 複数のパラメータを持つ **wildfly-core-eap** で定義されたメッセージでは、パラメータの順序が日本語と中国語で正しくありません。
この問題の修正は JBoss EAP 7.3.1 で予定されています。

例を以下に示します。

元の英語のテキスト:

```
Could not start app client <Parameter 1> as no main method was found on main class <Parameter 2>
```

現在の (正しい) 日本語訳:

メインクラス <Parameter 1> main メソッドが見つからなかったため、アプリケーションクライアント <Parameter 2> を起動できませんでした。

正しい日本語翻訳:

メインクラス <Parameter 2> main メソッドが見つからなかったため、アプリケーションクライアント <Parameter 1> を起動できませんでした。

Revised on 2021-03-31 13:26:02 +1000