



Red Hat JBoss Enterprise Application Platform 8.0

Red Hat JBoss Enterprise Application Platform 8.0 のリリースノート

Red Hat JBoss Enterprise Application Platform リリース 8.0 に関する重要な情報を記載したリリースノート

Red Hat JBoss Enterprise Application Platform 8.0 Red Hat JBoss Enterprise Application Platform 8.0 のリリースノート

Red Hat JBoss Enterprise Application Platform リリース 8.0 に関する重要な情報を記載したリリースノート

法律上の通知

Copyright © 2024 Red Hat, Inc.

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

このリリースノートには、Red Hat JBoss Enterprise Application Platform リリース 8.0 に関する重要な情報が記載されています。

目次

はじめに	4
JBOSS EAP ドキュメントへのフィードバック (英語のみ)	5
多様性を受け入れるオープンソースの強化	6
第1章 RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM 8.0 ドキュメントの読み方	7
第2章 サポートされる構成	8
OpenShift イメージ	8
Red Hat build of Keycloak SAML アダプターのサポート	8
MariaDB ドライバーのロギング依存関係	8
JBoss EAP 8.0 Operator	8
第3章 新機能および機能拡張	9
3.1. JAKARTA EE 10 のサポート	9
3.2. RED HAT INSIGHTS JAVA クライアント	10
3.3. 管理コンソール	10
3.4. 管理 CLI	14
3.5. セキュリティー	15
3.6. クラスタリング	19
3.7. JAKARTA EE	19
3.8. データソースのサブシステム	19
3.9. HIBERNATE	19
3.10. INFINISPAN	20
3.11. メッセージング	20
3.12. WEB サーバー (UNDERTOW)	20
3.13. EJB3 サブシステム	21
3.14. OPENSIFT	21
3.15. OPERATOR	23
3.16. クイックスタートおよび BOM	24
3.17. サーバー移行ツール	24
3.18. ACTIVEMQ ARTEMIS	24
3.19. JAKARTA FACES の実装	25
3.20. 高可用性	25
3.21. JBOSS-EAP-INSTALLATION-MANAGER	25
3.22. JBOSS-EAP-INSTALLATION-MANAGER の管理 CLI 統合	25
3.23. JBOSS-EAP-INSTALLATION-MANAGER の WEB コンソール統合	25
3.24. JBOSS EAP アプリケーションの移行	26
第4章 サポート対象外の機能、非推奨機能、廃止された機能	27
4.1. サポート対象外の機能	27
4.2. 非推奨の機能	27
4.3. 削除された機能	27
第5章 解決した問題	32
第6章 修正された CVE	33
第7章 既知の問題	34
7.1. INFINISPAN	34
7.2. データソース設定	35
7.3. サーバー管理	35
7.4. メッセージングフレームワーク	36

はじめに

このリリースノートには、Red Hat JBoss Enterprise Application Platform 8.0 に関する重要な情報が記載されています。

JBoss EAP ドキュメントへのフィードバック (英語のみ)

エラーを報告したり、ドキュメントを改善したりするには、Red Hat Jira アカウントにログインし、課題を送信してください。Red Hat Jira アカウントをお持ちでない場合は、アカウントを作成するように求められます。

手順

1. [このリンクをクリック](#) してチケットを作成します。
2. **Summary** に課題の簡単な説明を入力します。
3. **Description** に課題や機能拡張の詳細な説明を入力します。問題があるドキュメントのセクションへの URL を含めてください。
4. **Submit** をクリックすると、課題が作成され、適切なドキュメントチームに転送されます。

多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。多様性を受け入れる用語に変更する取り組みの詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) を参照してください。

第1章 RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM 8.0 ドキュメントの読み方

現在、Red Hat JBoss Enterprise Application Platform 8.0 ドキュメントのモダナイゼーションを進めています。さらにソリューション中心のドキュメントを作成できるよう取り組んでいます。

JBoss EAP 8.0 ドキュメントには、JBoss EAP 8.0 の新機能や拡張機能など、JBoss EAP 8.0 リリースに固有の内容が記載されています。JBoss EAP 8.0 で引き続きサポートされている以前のリリースの機能は、JBoss EAP 7.4 ドキュメントセットから参照できます。このドキュメントセットには、[Red Hat JBoss Enterprise Application Platform 7.4 製品ドキュメント](#) でアクセスできます。

JBoss EAP 8.0 ドキュメントは、以下の方法で使用することを推奨します。

1. JBoss EAP 8.0 リリースノート参照し、新機能、拡張機能、サポート対象外の機能、および削除された機能を確認します。
2. その他の JBoss EAP 8.0 ドキュメントセット参照し、新機能と拡張機能の詳細を確認します。
3. JBoss EAP 8.0 「移行ガイド」参照し、アプリケーションを JBoss EAP 8.0 に移行する方法の詳細を確認します。
4. JBoss EAP 8.0 で拡張されていない以前のリリースのサポート対象機能に関する情報が必要な場合は、[Red Hat JBoss Enterprise Application Platform 7.4 製品ドキュメント](#) の JBoss EAP 7.4 ドキュメントセット参照します。たとえば、JBoss EAP 7.4 ドキュメントセットでは開発ガイドや設定ガイドを利用できます。

第2章 サポートされる構成

Java 仮想マシン (JVM) と JBoss EAP 8.0 のサポート対象およびテスト済みの構成に関する詳細 (一般的に使用されるオペレーティングシステム、データベース、JMS ブローカーなど) は、Red Hat カスタマーポータルナレッジベース記事 [Red Hat JBoss Enterprise Application Platform \(EAP\) 8 でサポートされる構成](#) を参照してください。

含まれるモジュール、サポートされる標準、および Red Hat JBoss Enterprise Application Platform コンポーネントの詳細は、次のサポートドキュメントを参照してください。

- [Red Hat JBoss Enterprise Application Platform \(EAP\) 8 に含まれるモジュール](#)
- [Red Hat JBoss Enterprise Application Platform でサポートされる標準](#)
- [Red Hat JBoss Enterprise Application Platform コンポーネントの詳細](#)

OpenShift イメージ

ビルダーイメージとランタイムイメージは、Intel、IBM Systems Z および P、および ARM アーキテクチャー上の OpenJDK 17/RHEL 8 でサポートされます。

Red Hat build of Keycloak SAML アダプターのサポート

Keycloak SAML アダプター Galleon 機能パックを使用して、Red Hat build of Keycloak SAML アダプターをインストールできるようになりました。Red Hat build of Keycloak については、[Red Hat build of Keycloak の製品ページ](#) を参照してください。

MariaDB ドライバーのロギング依存関係

JBoss EAP 8.0 では、MariaDB ドライバーに **org.slf4j** モジュールへの依存関係が組み込まれており、ロギング機能が向上します。これにより、**slf4j** クラスが使用できない場合に発生する可能性のあるエラーが防止され、ログメッセージが適切に処理されるようになります。MariaDB ドライバーモジュールをデプロイするときは、この機能拡張を活用するために、**slf4j** 依存関係が利用可能であることを確認する必要があります。詳細は、[How to configure MariaDB driver 3.0+ as a JBoss Module in EAP 7 / 8](#) を参照してください。

JBoss EAP 8.0 Operator

JBoss EAP 8.0 は、EAP Operator をサポートするようになりました。JBoss EAP Operator を使用して、JBoss EAP 8.0 アプリケーションをデプロイできるようになりました。詳細は、[JBoss EAP Operator サポートポリシー](#) を参照してください。

第3章 新機能および機能拡張

JBoss EAP 8.0 では、以下の新機能および機能拡張が追加されました。

3.1. JAKARTA EE 10 のサポート

JBoss EAP 8 は Jakarta EE 10 のサポートを提供し、以下を含む Jakarta EE 10 Core Profile、Web Profile、および Full Platform 標準を実装します。

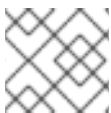
- Jakarta Activation 2.1
- Jakarta Annotations 2.1
- Jakarta Authentication 3.0
- Jakarta Authorization 2.1
- Jakarta Batch 2.1
- Jakarta Bean Validation 3.0
- Jakarta Concurrency 3.0
- Jakarta Connectors 2.1
- Jakarta Contexts and Dependency Injection 4.0
- Jakarta Debugging Support for Other Languages 2.0
- Jakarta Dependency Injection 2.0
- Jakarta Enterprise Beans 4.0
- Jakarta Enterprise Web Services 2.0
- Jakarta Expression Language 5.0
- Jakarta Interceptors 2.1
- Jakarta JSON Binding 3.0
- Jakarta JSON Processing 2.1
- Jakarta Mail 2.1
- Jakarta Messaging 3.1
- Jakarta Persistence 3.1
- Jakarta RESTful Web Services 3.1
- Jakarta Security 3.0
- Jakarta Server Faces 4.0
- Jakarta Server Pages 3.1

- Jakarta Servlet 6.0
- Jakarta SOAP with Attachments 1.3
- Jakarta Standard Tag Library 3.0
- Jakarta Transactions 2.0
- Jakarta WebSocket 2.1
- Jakarta XML Binding 4.0
- Jakarta XML Web Services 4.0

Jakarta EE 10 には、Jakarta EE 8 と比較して多くの変更点があります。詳細は、[JBoss EAP アプリケーションを Jakarta EE 8 から Jakarta EE 10 に移行する方法](#) を参照してください。

パッケージ名前空間の変更

すべての EE API で使用されるパッケージが **javax** から **jakarta** に変更されました。これは、Java EE が Eclipse Foundation に移行し、Jakarta EE が確立されたことに続くものです。



注記

この変更は、Java SE に含まれる **javax** パッケージには影響しません。

関連情報

- 詳細は、[javax から jakarta パッケージ名前空間への変更](#) を参照してください。

3.2. RED HAT INSIGHTS JAVA クライアント

JBoss EAP のバージョン 8.0 には、Red Hat Insights Java クライアントが含まれます。Red Hat Insights Java クライアントは、JBoss EAP が Red Hat Enterprise Linux (RHEL) にインストールされ、RHEL システムに Red Hat Insights クライアントがインストールされ、設定、登録されている場合にのみ有効になります。詳細は、[Red Hat Insights のクライアント設定ガイド](#) を参照してください。

Red Hat Insights dashboard for Runtimes は、[Red Hat Hybrid Cloud Console](#) の今後のリリースで利用可能になります。Red Hat Hybrid Cloud Console で利用可能な [RHEL ダッシュボード](#) と同様に、Runtimes ダッシュボードにはランタイムインストール、CVE の詳細、および JVM オプションの選択に役立つインベントリが表示されます。

環境変数 **RHT_INSIGHTS_JAVA_OPT_OUT** を **true** に設定すると、Red Hat Insights クライアントをオプトアウトできます。詳細は、ナレッジベースのアーティクル記事 [Red Hat Insights for Runtimes](#) を参照してください。

3.3. 管理コンソール

包摂的な言語、ラベルの変更

コード、ドキュメント、および Web プロパティ内の問題のある用語を置き換えるという Red Hat の取り組みの一環として、8.0 以降の JBoss EAP 管理コンソールでは、より多様性に配慮した用語とラベルが表示されるようになります。具体的には、管理コンソールのリソースアドレスとユーザーインターフェイス要素が次のように変更されています。

新しい用語	これまでの用語
プライマリー	マスター
セカンダリー	スレーブ
ブロックリスト	ブラックリスト
許可リスト	ホワイトリスト

応答メッセージへの定数 HTTP ヘッダーの追加、編集、および削除

JBoss EAP 8.0 管理コンソールで、定数 HTTP 応答ヘッダーを追加、編集、または削除できるようになりました。新しいパスおよびヘッダーを追加するには、サーバーページから **Constant Headers** を選択し、**Add** をクリックします。既存のパスヘッダーを編集または削除するには、ヘッダーを変更するパスを選択し、**Edit** または **Remove** をクリックします。

処理されたメッセージの Java Message Service ブリッジ統計情報の表示

メッセージブリッジは、ソースキューまたはトピックからメッセージを消費し、通常は別のサーバー上にあるターゲットキューまたはトピックに送信します。ブリッジは、あるクラスターから別のクラスターにメッセージを送信することもできます。Java Message Service (JMS) ブリッジは、ブリッジが処理したメッセージに関する統計情報を提供します。具体的には、次のデータを収集します。

- 正常にコミットされたメッセージの数 (メッセージ数)
- 中断されたメッセージの数 (中断されたメッセージ)

今回の更新により、JBoss EAP 8.0 管理コンソールの Runtime セクションに、これらの統計を表示するための新しい **JMS Bridge** 列が追加されました。この新機能は `/subsystem=messaging-activemq/jms-bridge=*` リソースに影響することに注意してください。

拡張監査ログの設定

JBoss EAP 8.0 管理コンソールでは、`/subsystem=elytron/syslog-audit-log=*` リソースで次の 2 つの追加の監査ロギング属性を設定できます。

- **syslog-format**
監査ログメッセージの形式を定義します。サポートされている値は **RFC3164** および **RFC5424** です ("RFC" は "request for comments" の略)。
- **reconnect-attempts**
エンドポイントを閉じる前に JBoss EAP が syslog サーバーへの接続を試行して失敗する最大回数を定義します。

/deployment サブリソースに include-runtime=true が必要

Red Hat JBoss Enterprise Application Platform 8.0 では、`/deployment` のサブモデルがランタイムに変更されました。`/deployment` サブリソースを使用する管理操作の場合は、**include-runtime=true** を追加する必要があります。

一時停止モードでのサーバーの始動

JBoss EAP 8.0 管理コンソールを使用して、サーバーを一時停止モードで起動できるようになりました。次のドロップダウンメニューで、新しい **Start in suspended mode** オプションを選択します。

- Runtime > Topology

- Runtime > Server Groups
- Runtime > Server Groups > Server
- Runtime > Host > Server

certificate-authority-account リソースの認証局属性の設定

JBoss EAP 8.0 では、**certificate-authority-account** Elytron リソースに任意の認証局を使用できます。以前は、JBoss EAP は Let's Encrypt 認証局のみをサポートしており、**certificate-authority** 属性は設定できませんでした。

今回の更新では、JBoss EAP 管理コンソールを開き、**Configuration > Subsystems > Security > Other Settings > Other Settings > Certificate Authority** をクリックして、認証局を追加、設定、または削除できます。そこから **Add** をクリックして、新しい認証局を追加します。すでにあるものを変更するには、それを選択して **Edit** をクリックします。認証局を削除するには、認証局を選択して **Remove** をクリックします。

OCSP を Elytron トラストマネージャーとして設定

JBoss EAP 8.0 では、Online Certificate Status Protocol (OCSP) を Elytron **undertow** サブシステムのトラストマネージャーとして設定できます。これまで JBoss EAP では、トラストマネージャーとして証明書失効リスト (CRL) のみをサポートしていました。

今回の更新により、JBoss EAP 管理コンソールを開き、**Configuration > Subsystems > Elytron > Other Settings > SSL > Trust Manager** をクリックすることで、OCSP をトラストマネージャーとして設定できるようになりました。次に、トラストマネージャーを選択または作成し、Trust Manager ウィンドウで **OCSP** タブを選択して **Add** をクリックします。

Java Message Service トピックの一時停止

JBoss EAP 8.0 管理コンソールから、**Runtime > Messaging > Server > Server Name > Destination** に移動して、Java Message Service (JMS) トピックを選択して一時停止できるようになりました。関連するメッセージの問題に対処した後、一時停止したトピックを再開することもできます。以前は、JMS はすべてのアクティブなサブスクリバに、中断する方法なしでメッセージを送信していました。

サーバーステータスプレビューに追加された非ヒープメモリ使用量

JBoss EAP 8.0 では、サーバーステータスプレビューで確認できるサーバーのメモリ消費情報が増えました。以前は、プレビューにはヒープメモリの使用量 (**Used** および **Committed**) のみが表示されていました。今回の更新により、ヒープ以外のメモリ使用量についても同じ情報が表示されます。

データソースを追加または更新するときに認証情報ストアのパスワードを自動的に追加または更新します

JBoss EAP 8.0 以降、管理コンソールからデータソースを作成すると、そのデータソースのパスワードを認証情報ストアに自動的に追加できます。管理コンソールから **Configuration > Subsystems > Datasources** を選択し、**Add** をクリックして新しいデータソースを追加します。次に、新しいデータソースのパスワードを保存する認証情報ストア名、認証情報の別名、および使用するプレーンテキストのパスワードを入力します。既存のデータソースを変更するには、それを選択して **Edit** をクリックします。

Elytron リソースの作成、読み取り、更新、および削除

JBoss EAP 8.0 管理コンソールから、次の 4 つのエビデンスデコーダーのいずれかを作成、読み取り、更新、または削除できるようになりました。

- Aggregate Evidence Decoder
- Custom Evidence Decoder
- X500 Subject Evidence Decoder

- X509 Subject Alt Name Evidence Decoder

これらのアクションのいずれかを実行するには、**Configuration > Subsystems > Security > Mappers & Decoders > Evidence Decoder** に移動します。

デプロイハッシュ値の表示

JBoss EAP 8.0 管理コンソールは、デプロイメントプレビューでデプロイメントハッシュ値を表示できるようになりました。これは、デプロイメントが有効で成功したかどうかを一目で判断できることを意味します。

EJB 3 サブシステムでのインターセプターの追加および設定

JBoss EAP 8.0 管理コンソールから、システム全体のサーバー側インターセプターを **ejb3** サブシステムに追加および設定できるようになりました。コンソールから、**Configuration > EJB > Container** を選択して、追加または変更を行います。

Infinispan 分散 Web セッションアフィニティの設定

JBoss EAP 8.0 では、**distributable-web** サブシステムで、分散 Web セッションのアフィニティ (ロードバランサーの "スティッキネス") をより詳細に制御できるようになりました。セッションアフィニティを **Primary-owner** のデフォルト以外に変更するには、管理コンソールで **Configuration > Distributable Web > View > Infinispan Session** に移動します。次に、セッションを選択し、**アフィニティ** を選択して変更を加えます。アフィニティオプションには、次のものが含まれるようになりました。

- Local
- None
- Primary-owner
- Ranked

以前は、利用可能なアフィニティは **Primary-owner** のみでした。

EE サブシステムでのグローバルディレクトリーの設定

JBoss EAP 8.0 管理コンソールを使用して、新しい **ee** サブシステムリソース **/subsystem=ee/global-directory=*** を設定できるようになりました。グローバルディレクトリーを使用すると、ディレクトリーの内容を表示せずに、デプロイメントクラスパスに内容を追加できます。グローバルディレクトリーリソースを設定するには、**Configuration > Subsystems > EE > Globals** に移動します。

Elytron での暗号スイートの設定

JBoss EAP 8.0 管理コンソールでは、**cipher-suite-names** 属性を使用して TLS 1.3 暗号スイートを有効にし、ネットワーク接続を保護できるようになりました。具体的には、以下の **elytron** サブシステムリソースを設定できるようになりました。

- **/subsystem=elytron/client-ssl-context=***
- **/subsystem=elytron/server-ssl-context=***

管理コンソールから **/subsystem=elytron/client-ssl-context=*** リソースの **cipher-suite-names** 属性を設定するには、**Configuration > Subsystems > Security > Other Settings > SSL > Client SSL Context** に移動します。

管理コンソールから **/subsystem=elytron/server-ssl-context=*** リソースの **cipher-suite-names** 属性を設定するには、**Configuration > Subsystems > Security > Other Settings > SSL > Server SSL Context** に移動します。

OIDC によるアプリケーションと管理コンソールの保護

JBoss EAP 8.0 では、管理コンソールから OpenID Connect (OIDC) を使用して、JBoss EAP にデプロイされたアプリケーションと、JBoss EAP 管理コンソールを保護できます。JBoss EAP 8.0 は、**elytron-oidc-client** サブシステムにより OpenID Connect (OIDC) をネイティブでサポートします。

管理コンソールから **elytron-oidc-client** サブシステムを設定するには、**Configuration > Subsystems > Elytron OIDC Client** に移動します。

JBoss EAP にデプロイされたアプリケーションを保護するには、以下のリソースを設定します。

- **provider**
- **secure-deployment**

詳細は、[JBoss EAP での Single Sign-On の使用ガイドの OIDC によるアプリケーションの保護](#) を参照してください。

JBoss EAP 管理インターフェイスを保護するには、以下のリソースを設定します。

- **provider**
- **secure-deployment**
- **secure-server**

また、管理コンソールを OIDC で保護する場合は、**Access Control** に移動して **Enable RBAC** をクリックすることで、管理コンソールのロールベースのアクセス制御 (RBAC) を設定できます。

詳細は、[JBoss EAP での Single Sign-On の使用ガイドの OpenID プロバイダーによる JBoss EAP 管理コンソールの保護](#) を参照してください。



注記

realm リソースを使用すると、Red Hat build of Keycloak レルムを設定できます。これは便宜上提供されています。keycloak クライアントアダプターの設定をコピーして、**realm** リソース設定で使用できます。しかし、それよりも **provider** リソースを使用することが推奨されます。

3.4. 管理 CLI

アプリケーションデプロイ時の Web コンテキストの登録

管理コマンドラインインターフェイス (CLI) から **deployment apply-file** コマンドを使用して、アプリケーションをスタンドアロンサーバーまたはマネージドドメインにデプロイできます。

アプリケーションをスタンドアロンサーバーにデプロイする

```
deployment deploy-file /path/to/test-application.war
```

マネージドドメイン内のすべてのサーバーグループにアプリケーションをデプロイする

```
deployment deploy-file /path/to/test-application.war --all-server-groups
```

マネージドドメイン内の特定のサーバーグループにアプリケーションをデプロイする

```
deployment deploy-file /path/to/test-application.war --server-groups=main-server-group,other-server-group
```

前述の例では、**runtime-name** 属性のデフォルト値は **test-application.war** です。

--runtime-name オプションで **runtime-name** 属性を指定する場合は、名前に **.war** 拡張子を含める必要があります。そうしないと、Web コンテキストが JBoss EAP によって登録されません。以下に例を示します。

```
--runtime-name=my-application.war
```

3.5. セキュリティー

elytron サブシステムの JAAS レルム

JBoss EAP 8.0 では、レガシーセキュリティーサブシステムが削除されました。**elytron** サブシステムでカスタムログインモジュールを引き続き使用するには、新しい Java Authentication and Authorization Service (JAAS) セキュリティーレルム **jaas-realm** を使用します。



注記

jaas-realm は、JAAS 互換のログインモジュールのみをサポートします。JAAS の詳細は、[Java Authentication and Authorization Service \(JAAS\) Reference Guide](#) を参照してください。

jaas-realm は、PicketBox API を拡張する、またはそれに依存するカスタムログインモジュールをサポートしません。

elytron サブシステムは **jaas-realm** を提供しますが、サブシステムが提供する他の既存セキュリティーレルムの使用を推奨します。これには、**jdbc-realm**、**ldap-realm**、**token-realm** などが含まれます。**aggregate-realm**、**distributed-realm**、または **failover-realm** を設定すると、異なるセキュリティーレルムを組み合わせることもできます。そのいずれも目的に適さない場合は、カスタムセキュリティーレルムを実装し、カスタムログインモジュールの代わりに使用してください。

以下の場合、カスタムセキュリティーレルムを実装する代わりに **jaas-realm** を使用する必要があります。

- レガシーセキュリティーから **elytron** サブシステムに移行しようとしており、すでにカスタムログインモジュールが実装されている。
- 他のアプリケーションサーバーから JBoss EAP に移行し、ログインモジュールがすでに実装されている。
- 複数のログインモジュールを、それらのログインモジュールに提供されているさまざまなフラグとオプションと組み合わせる必要がある。これらのフラグとオプションは、**elytron** サブシステムで提供されているセキュリティーレルムに対して設定できない場合があります。

詳細は、[複数のアイデンティティストアを使用したアプリケーションと管理インターフェイスの保護ガイドの AAS レルムの作成](#) を参照してください。

Elytron および Elytron クライアントでの複数の証明書失効リストの設定

複数の認証局 (CA) を使用する場合、**elytron** サブシステムおよび WildFly Elytron クライアントで複数の証明書失効リスト (CRL) を設定できるようになりました。**trust-manager** の **certificate-revocation-lists** 属性で使用する CRL の一覧を指定できます。

詳細は、JBoss EAP の SSL/TLS 設定ガイドの [Elytron における証明書失効チェックの設定](#) を参照してください。

Keycloak SAML アダプター機能パック

Keycloak SAML アダプターのアーカイブディストリビューションは、JBoss EAP では提供されなくなりました。代わりに、Keycloak SAML アダプター機能パックを使用して、**keycloak-saml** サブシステムおよび関連設定をインストールできます。

Keycloak SAML アダプター機能パックは、ユースケースに応じてインストールできる次のレイヤーを提供します。

- **keycloak-saml**
- **keycloak-client-saml**
- **keycloak-client-saml-ejb**

詳細は、[JBoss EAP での Single Sign-On の使用ガイド](#) を参照してください。

ネイティブ OpenID Connect クライアント

JBoss EAP は、**elytron-oidc-client** サブシステムにより OpenID Connect (OIDC) をネイティブでサポートします。そのため、Red Hat build of Keycloak クライアントアダプターは、このリリースでは提供されません。**elytron-oidc-client** サブシステムは、Relying Party (RP) として機能します。**elytron-oidc-client** サブシステムは、ベアラーのみの認証をサポートするほか、マルチテナンシーのサポートを提供します。マルチテナンシーのサポートを使用すると、たとえば、複数の Red Hat build of Keycloak レルムからのユーザーをアプリケーションに対して認証できます。



注記

JBoss EAP のネイティブ OIDC クライアントは、RP が開始するログアウトをサポートしません。

elytron-oidc-client サブシステムを使用すると、JBoss EAP および JBoss EAP 管理コンソールにデプロイされたアプリケーションを OIDC で保護できます。

さらに、次のどちらの場合も、OIDC プロバイダーから取得したセキュリティーアイデンティティーをサーブレットから Jakarta Enterprise Bean に伝播できます。

- サーブレットと Jakarta Enterprise Bean が同じデプロイメント内にある。
- サーブレットと Jakarta Enterprise Bean が異なるデプロイメント内にある。

詳細は、[JBoss EAP での Single Sign-On の使用ガイド](#) を参照してください。

ハッシュ値を持ったパスワードの新しい hash-encoding 属性および hash-charset 属性

hash-charset 属性と **hash-encoding** 属性を使用して、**elytron** サブシステムセキュリティーレルムに保管されるハッシュ化されたパスワードの文字セットと文字列形式を指定できるようになりました。デフォルトの **hash-charset** 値は **UTF-8** です。**hash-encoding** 値を **base64** または **hex** に設定できます。**base64** は、**hex** がデフォルトである **properties-realm** を除くすべてのレルムのデフォルトです。

新しい属性は、次のセキュリティーレルムに含まれています。

- **filesystem-realm**
- **jdbc-realm**

- `ldap-realm`
- `properties-realm`

詳細は、[アイデンティティストアを使用したアプリケーションと管理インターフェイスの保護](#) ガイドを参照してください。

Elytron ファイルベースの監査ログの新しい `encoding` 属性

`encoding` 属性を使用して、Elytron でファイルベースの監査ログのエンコーディングを指定できるようになりました。デフォルト値は **UTF-8** です。以下の値が可能です。

- **UTF-8**
- **UTF-16BE**
- **UTF-16LE**
- **UTF-16**
- **US-ASCII**
- **ISO-8859-1**

詳細は、[アイデンティティストアを使用したアプリケーションと管理インターフェイスの保護](#) ガイドの [Elytron 監査ロギング](#) を参照してください。

SSLv2Hello

JBoss EAP 8.0 Beta 以降、**elytron** サブシステムで **server-ssl-context** および **client-ssl-context** に **SSLv2Hello** プロトコルを指定できるようになりました。



警告

- 後者の目的は、接続されたサーバーがサポートする暗号化プロトコルを決定することであるため、**SSLv2Hello** を設定する場合は、別の暗号化プロトコルを設定する必要があります。
- **IBM JDK** は、クライアントで **SSLv2Hello** をサポートしていませんが、サーバー側の接続は常にこのプロトコルを受け入れます。

filesystem-realmの更新

セキュリティを強化するために、**filesystem-realm** のアイデンティティに関連付けられた明確なパスワード、ハッシュ化されたパスワード、および属性を暗号化できるようになりました。これは、以下の2つの方法で実行できます。

- **add** 操作で秘密鍵を参照して、暗号化された **filesystem-realm** を作成します。
- WildFly Elytron Tool で新しい **filesystem-realm-encrypt** コマンドを使用して、既存の **filesystem-realm** を暗号化します。

また、**filesystem-realm** の整合性チェックを有効にして、最後に承認された書き込み以降に **filesystem-realm** のアイデンティティが改ざんされないようにすることもできます。これは、**add** 操作を使用して **filesystem-realm** を作成する際にキーペアを参照して実行できます。Elytron は、キーペ

アを使用してアイデンティティファイルの署名を生成します。整合性チェックは、アイデンティティファイルが読み込まれるたびに実行します。

詳細は、[アイデンティティストアを使用したアプリケーションと管理インターフェイスの保護ガイド](#)の [Elytron のファイルシステムレルム](#) を参照してください。

distributed-realm の更新

新しい属性 **ignore-unavailable-realms** を **true** に設定することで、アイデンティティストアへの接続が失敗した場合でも、参照先のセキュリティレルムの検索を継続するように **distributed-realm** を設定できるようになりました。

デフォルトでは、アイデンティティが一致する前にアイデンティティストアへの接続が失敗した場合、以前と同様に認証が例外 **RealmUnavailableException** で失敗します。

ignore-unavailable-realms を **true** に設定すると、クエリーされたレルムのいずれかが使用できない場合に **SecurityEvent** が発行されます。この動作は、**emit-events** を **false** に設定することで設定できません。

詳細は、[複数のアイデンティティストアを使用したアプリケーションと管理インターフェイスの保護ガイド](#)の次のリソースを参照してください。

- [Elytron での分散レルム](#)
- [distributed-realm 属性](#)

Artemis の SSLContexts に対する Elytron サポートの提供

JBoss EAP 8 では、Messaging サブシステムで **SSLContext** 変数をインスタンス化するための Elytron サポートが提供されます。この機能により、Elytron がこの変数をインスタンス化するとき、複数の場所で **SSLContext** を設定する手間が省けます。SSLContext のコネクターは、クライアントの JBoss EAP サーバーの **elytron** サブシステムで定義する必要があります。そのため、スタンドアロンのメッセージングクライアントアプリケーションからコネクターを定義することはできません。

Elytron クライアントの新しい Java セキュリティプロバイダー

Elytron クライアントが、Java セキュリティプロバイダー

org.wildfly.security.auth.client.WildFlyElytronClientDefaultSSLContextProvider を提供するようになりました。これを使用すると、Java 仮想マシン (JVM) 全体のデフォルトの **SSLContext** を登録できます。

十分に高い優先度を使用して JVM にプロバイダーを登録すると、**SSLContext.getDefault()** メソッドを使用するすべてのクライアントライブラリーが、Elytron クライアント設定でデフォルトとして設定されている SSL コンテキストのインスタンスを取得します。このようにして、Elytron API と直接やり取りすることなく、Elytron クライアントの SSL コンテキスト設定を利用できます。

詳細は、[JBoss EAP における SSL/TLS の設定ガイド](#)の [Elytron クライアントのデフォルト SSLcontext セキュリティプロバイダーを JBoss EAP クライアントで使用する](#) を参照してください。

Elytron からカスタムプリンシパルを取得する機能

JBoss EAP 8.0 では、Elytron からカスタムプリンシパルを取得できるようになりました。以前は、Elytron では認証用としてプリンシパルが **NamePrincipal** のインスタンスである必要がありました。現在の **SecurityDomain** から取得した **SecurityIdentity** を使用し、**SecurityIdentity** 属性を使用してレルムから情報を取得することができましたが、**jakarta.security.enterprise.SecurityContext.getCallerPrincipal()** などのより汎用的で標準化されたメソッドではなく、**SecurityDomain** および **SecurityIdentity** に依存する必要がありました。

Elytron の使用時に、**getCallerPrincipal()** メソッドからカスタムプリンシパルを取得できるようになりました。レガシーセキュリティを使用したアプリケーションコードが、カスタムプリンシパルを **getCallerPrincipal()** メソッドから取得することに依存する場合は、コードの変更を必要とせずにアプ

リケーションを移行できます。

3.6. クラスタリング

ProtoStream を使用した Web セッションレプリケーションの設定

JBoss EAP 8.0 では、JBoss Marshalling の代わりに ProtoStream を使用して Web セッションレプリケーションを設定できるようになりました。

[How to configure web session replication to use ProtoStream instead of JBoss Marshalling in JBoss EAP 8.0](#) を参照してください。

別のノードからのバッチジョブの実行停止

JBoss EAP 8.0 の別のクラスター化されたノードからのバッチジョブの実行を停止できるようになりました。詳細は、[JBoss EAP 8.0 で同じジョブリポジトリを共有するノードのクラスタリングでバッチ処理 JBeret を使用する](#) を参照してください。

3.7. JAKARTA EE

Jakarta EE Core Profile

Jakarta EE 10 Core Profile が JBoss EAP 8.0 で利用できるようになりました。Core Profile は、マイクロサービスやクラウドサービスなどの小規模なランタイムに適した Jakarta EE 仕様を提供する小型軽量のプロファイルです。Jakarta EE 10 Core Profile は、Galleon プロビジョニングレイヤー **ee-core-profile-server** として利用できます。

Core Profile Galleon レイヤーの詳細は、[JBoss EAP for OpenShift の機能調整: ベースレイヤー](#) を参照してください。

3.8. データソースのサブシステム

データソースのカスタムの **exception-sorter** または **valid-connection-checker** の設定

JBoss モジュールを使用して、データソースのカスタム **exception-sorter** または **valid-connection-checker** を設定できるようになりました。

[How to configure a custom exception-sorter or valid-connection-checker for a datasource in JBoss EAP 8](#) を参照してください。

JBoss EAP 8.0 の **eap-datasources-galleon-pack** のサポート

eap-datasources-galleon-pack Galleon 機能パックを使用して、データベースに接続できる JBoss EAP 8.0 サーバーをプロビジョニングできるようになりました。

3.9. HIBERNATE

Hibernate Search 6 による Hibernate Search 5 API の置き換え

JBoss EAP 8.0 では、Hibernate Search 5 API が削除され、Hibernate Search 6 API に置き換えられました。

削除された機能のリストを確認するには、[Hibernate Search 5 APIs Deprecated in JBoss EAP 7.4 and removed in EAP 8.0](#) を参照してください。



注記

Hibernate Search 6 API には、Hibernate Search 5 API との **下位互換性はありません**。アプリケーションを Hibernate Search 6 に移行する必要があります。

JBoss EAP 8.0 に含まれる Hibernate Search 6 の最新バージョンは 6.2 です。Hibernate Search 5 から移行する場合は、バージョン 6.0、6.1、**および** 6.2 への移行を検討してください。

詳細は、次の移行ガイドを参照してください。

- アプリケーションを Hibernate Search 5 から移行するには、[Hibernate Search 6.0 の移行ガイド](#) を参照してください。
- アプリケーションを Hibernate Search 6.0 から 6.1 に移行するには、[Hibernate Search 6.1 の移行ガイド](#) を参照してください。
- アプリケーションを Hibernate Search 6.1 から 6.2 に移行するには、[Hibernate Search 6.2 の移行ガイド](#) を参照してください。



注記

Hibernate Search 6.2 は Hibernate ORM 6.2 と互換性があります。詳細は、Hibernate Search 6.2 リファレンスドキュメントの [Hibernate ORM 6](#) セクションを参照してください。

Hibernate Search 6 による Elasticsearch のサポート

JBoss EAP 8.0 は、Hibernate Search 6 で Elasticsearch バックエンドを使用してリモート Elasticsearch クラスターまたは OpenSearch クラスターにデータのインデックスを作成するためのサポートも提供します。

使用可能な Hibernate Search アーキテクチャーとバックエンドのリストを確認するには、Hibernate Search 6.2 リファレンスドキュメントの [Table 2. Comparison of architectures](#) を参照してください。

Hibernate Search 6 の設定の詳細は、「WildFly Developer guide」の [Using Hibernate Search](#) を参照してください。

3.10. INFINISPAN

Infinispan の分散クエリー、カウンター、およびロック API と CDI モジュールのサポート

JBoss EAP 8.0 では、分散クエリー、カウンター、およびロックに Infinispan API を使用できるようになりました。

JBoss EAP 8.0 では、キャッシュの作成と注入のために Infinispan CDI モジュールも使用できます。

詳細は、[EAP 8 now supports Infinispan query, counters, locks, and CDI](#) を参照してください。

3.11. メッセージング

新しい Galleon レイヤーの追加

新しい Galleon レイヤーが追加され、組み込み ActiveMQ Artemis ブローカーと Jakarta Messaging Service (JMS) 統合のサポートが提供されるようになりました。詳細は、「移行ガイド」の [組み込みブローカーメッセージング用の Galleon レイヤー](#) セクションを参照してください。

3.12. WEB サーバー (UNDERTOW)

Web リクエストアフィニティー用の Cookie の設定

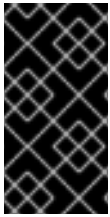
アドレス `/subsystem=undertow/servlet-container=default/setting=affinity-cookie` にある `affinity-cookie` リソースを使用して、ロードバランサーのセッションアフィニティー情報を保存する別の Cookie を設定できるようになりました。

詳細は、Red Hat ナレッジベースソリューションの [How to configure the affinity-cookie and session-cookie in JBoss EAP 8](#) を参照してください。

3.13. EJB3 サブシステム

JBoss EAP 8.0 サーバーと JBoss EAP 7 および JBoss EAP 6 の相互運用性

JBoss EAP 8.0 では、JBoss EAP 8.0 と古いバージョンの JBoss EAP サーバー間の相互運用性を有効にすることができます。JBoss EAP は、API クラスが **jakarta** パッケージ名前空間を使用する Jakarta EE 10 をサポートします。ただし、古いバージョンの JBoss EAP は **javax** パッケージの名前空間を使用します。



重要

- サポートされる旧バージョンは JBoss EAP 6 および JBoss EAP 7 です。
- JBoss EAP 6 と JBoss EAP 7 間の相互運用性は、両方のサーバーが **javax** パッケージの名前空間をサポートしているため、この問題の影響を受けません。

JBoss EAP 8.0 と古いバージョンの JBoss EAP の間の相互運用性を有効にする方法の詳細には、[相互運用性を有効にする方法](#) を参照してください。

Infinispan ベースの分散タイマー

JBoss EAP 8.0 では、Infinispan ベースの分散タイマーを使用してクラスター内で永続的な Jakarta Enterprise Bean タイマーをスケジュールできるようになりました。このタイマーは大規模なクラスターにスケールできます。詳細は、[EAP 8 - how to configure Infinispan based distributed timers](#) を参照してください。

配布可能な EJB サブシステム

distributable-ejb サブシステムを使用して、次のような **ejb3** サブシステム機能に必要なクラスタリング抽象化プロバイダーを設定します。

- ステートフルセッション Bean (SFSB) キャッシュファクトリー
- EJB クライアントアプリケーションのクライアントマッピングレジストリー
- 分散 EJB タイマー

現在、これらのプロバイダーをシステム全体のレベルで定義できます。**ejb3** サブシステムをカスタマイズして、デプロイメント固有のプロバイダーを有効にする機能を開発する予定です。詳細は、[What is the distributable-ejb subsystem in EAP 8](#) を参照してください。

3.14. OPENSIFT

JBoss EAP 8.0 の Red Hat build of Keycloak SAML のサポート

JBoss EAP 8.0 の Source-to-Image (S2I) イメージでの Red Hat build of Keycloak SAML アダプターの使用は、アダプターのリリース時にサポートされる予定です。詳細は、[OpenShift, SSO SAML support for EAP 8](#) を参照してください。

Maven プラグインを使用した JBoss EAP サーバーのプロビジョニング

OpenShift で JBoss EAP Maven プラグインを使用して、以下を実行できるようになりました。

- Galleon を使用してトリミングされたサーバーをプロビジョニングします。
- プロビジョニングされたサーバーにアプリケーションをインストールします。

- JBoss EAP 管理 CLI を使用してサーバー設定を調整します。
- **keystore** ファイルなどの追加ファイルをサーバーインストールにパッケージ化します。
- プラグインを JBoss EAP 8.0 source-to-image アプリケーションビルドに統合します。

詳細は、[Maven プラグインを使用した JBoss EAP サーバーのプロビジョニング](#) を参照してください。

JBoss EAP source-to-image の OpenID Connect サポート

以前に必要な Red Hat build of Keycloak クライアントアダプターをインストールする代わりに、新しい **elytron-oidc-client** サブシステムを使用して、JBoss EAP にデプロイされたアプリケーションを OpenID Connect (OIDC) で保護できるようになりました。環境変数を使用して **elytron-oidc-client** サブシステムを設定し、OIDC でアプリケーションを保護できます。Red Hat build of Keycloak クライアントアダプターは、このリリースでは提供されません。詳細は、[OpenID Connect を使用して OpenShift 上の JBoss EAP アプリケーションを保護する](#) を参照してください。

Source-to-Image を使用したアプリケーションイメージのビルド

JBoss EAP 8.0 では、インストール済みサーバーが Source-to-Image (S2I) ビルダーイメージから削除されました。S2I ビルドフェーズ中にサーバーをプロビジョニングするために、Galeon 機能パックとレイヤーが使用されるようになりました。サーバーをプロビジョニングするには、アプリケーションの **pom.xml** ファイルに JBoss EAP Maven プラグインを含めて設定します。詳細は、[OpenShift で source-to-image を使用したアプリケーションイメージのビルド](#) を参照してください。

環境変数による管理属性のオーバーライド

JBoss EAP サーバー設定をサーバー環境により簡単に適応させるために、環境変数を使用して、設定ファイルを編集せずに管理属性の値をオーバーライドできます。**LIST**、**OBJECT**、または **PROPERTY** 型の管理属性はオーバーライドできません。JBoss EAP 8.0 の OpenShift ランタイムイメージでは、この機能がデフォルトで有効になっています。詳細は、[環境変数を使用した管理属性のオーバーライド](#) を参照してください。

管理モデル式を解決するための環境変数チェック

JBoss EAP は、管理モデル式を解決する際の環境変数チェックをサポートするようになりました。以前のバージョンの JBoss EAP では、JBoss EAP サーバーは管理式の Java システムプロパティーのみをチェックしていました。現バージョンのサーバーは、関連する環境変数とシステムプロパティーをチェックします。両方使用した場合、JBoss EAP は環境変数ではなく Java システムプロパティーを使用して管理モデル式を解決します。環境変数を使用して管理モデル式を解決する方法の詳細は、[環境変数とモデル式の解決の使用](#) を参照してください。

Maven の互換性

Maven バージョン 3.8.5 以前には、3.3.2 より前のバージョンの Apache Maven WAR プラグインが含まれています。これにより、**eap-maven-plugin** でパッケージ化エラーが発生します。この問題を解決するには、Maven バージョン 3.8.6 以降にアップグレードする必要があります。あるいは、**maven-war-plugin** 依存関係 (バージョン 3.3.2 以降) をアプリケーション **pom.xml** に追加することもできます。

ノード命名の機能拡張

jboss.node.name システムプロパティーの値が、Pod ホスト名から生成され、**JBOSS_NODE_NAME** 環境変数を使用してカスタマイズできるようになりました。このシステムプロパティーは、トランザクション ID として機能しなくなりました。また、以前のバージョンの JBoss EAP に存在した 23 文字の長さ制限がなくなりました。

ただし、JBoss EAP 8.0 では、新しいシステムプロパティー **jboss.tx.node.id** も、Pod ホスト名から生成され、**JBOSS_NODE_NAME** 環境変数を使用してカスタマイズできます。このシステムプロパティーは、長さが 23 文字に制限され、トランザクション ID として機能します。

JBoss EAP 8.0 イメージでの Java オプションの変更

JVM は、JBoss EAP 8.0 イメージのメモリーと CPU 制限、およびガベージコレクター設定を自動的に

調整します。-Xms および -Xmx オプションを計算する代わりに、イメージは -XX:InitialRAMPercentage および -XX:MaxRAMPercentage オプションを使用して、同じ機能を動的に実現します。CONTAINER_CORE_LIMIT および JAVA_CORE_LIMIT が削除されました。さらに、-XX:ParallelGCThreads, -Djava.util.concurrent.ForkJoinPool.common.parallelism、 および -XX:CICompilerCount が使用されなくなりました。

OpenShift へのサードパーティーアプリケーションのデプロイ

JBoss EAP 8.0 では、コンパイルされた WAR ファイルまたは EAR アーカイブを使用して、OpenShift デプロイメント用のアプリケーションイメージを作成できます。Dockerfile を使用すると、オペレーティングシステム、Java、および JBoss EAP コンポーネントを含む完全なランタイムスタックで、これらのアーカイブを JBoss EAP サーバーにデプロイできます。Source-to-Image (S2I) に依存せずにアプリケーションイメージを作成できます。

OpenShift 上の JBoss EAP 8.0 サーバーインストールで除外されたファイル

OpenShift に JBoss EAP 8.0 サーバーをインストールする場合、以下のファイルは必要なく、意図的に除外されます。

- bin/appclient.sh
- bin/wsprovide.sh
- bin/wsconsume.sh
- bin/jconsole.sh
- bin/client

3.15. OPERATOR

JBoss EAP 8.0 Operator

JBoss EAP 8.0 Operator は間もなく利用可能になります。JBoss EAP Operator が利用可能になると、Operator を使用して JBoss EAP 8.0 アプリケーションをデプロイできます。詳細は、[JBoss EAP Operator Support Policy](#) を参照してください。

JBoss EAP 8.0 Operator を使用した Health Probe 設定の強化

JBoss EAP 8.0 Operator では、ヘルスプローブの設定オプションが改善され、JBoss EAP 8.0 と JBoss EAP 7.4 イメージ間のプローブのカスタマイズと互換性が向上しました。この機能拡張により、両方のイメージ間の相互運用性が確実となり、プローブは実行方法を柔軟に調整できます。

JBoss EAP 8.0 インスタンスでの主な改善:

- JBoss EAP 8.0 および JBoss EAP 7 ベースのイメージと連携する機能
- **LivenessProbe**、**ReadinessProbe**、および **StartupProbes** を設定する機能。

Startup Probe の設定例:

```
apiVersion: wildfly.org/v1alpha1
kind: WildFlyServer
metadata:
  name: ...
spec:
  applicationImage: '...'
  livenessProbe:
    httpGet:
```

```

path: /health/live
port: 9990
scheme: HTTP
initialDelaySeconds: 30
readinessProbe:
  httpGet:
    path: /health/ready
    port: 9990
    scheme: HTTP
    initialDelaySeconds: 10
replicas: 1
startupProbe:
  httpGet:
    path: /health/started
    port: 9990
    scheme: HTTP
    initialDelaySeconds: 60

```



注記

デフォルトでは、JBoss EAP 8.0 アプリケーションはシェルプローブを保持し、JBoss EAP 7 ベースのアプリケーションの後方互換性を確保します。

3.16. クイックスタートおよび BOM

サポート対象の EAP 8 クイックスタート

サポート対象の JBoss EAP 8 クイックスタートは、すべて [jboss-eap-quickstarts](#) にあります。

新しい JBoss EAP BOM for Maven

JBoss EAP BOM は、Maven プロジェクトのビルドまたはテストに必要な JBoss EAP 依存関係のバージョンを指定する Maven BOM ファイルを提供します。さらに Jakarta EE 10 BOM は、Hibernate、RESTasy、Infinispan や Client BOM などのプロプライエタリーコンポーネントなどの関連フレームワークの依存関係管理機能を提供します。

3.17. サーバー移行ツール

JBoss EAP Server Migration Tool

Server Migration Tool はスタンドアロンの移行ツールになり、JBoss EAP 8.0 には含まれなくなりました。この移行ツールは個別にダウンロードできます。

3.18. ACTIVEMQ ARTEMIS

ActiveMQ サーバーでのブリッジ追加の失敗

JBoss EAP 7 では、ソースキューを作成する前に、**messaging-activemq** サブシステムに Java Message Service (JMS) ブリッジを作成できました。ブリッジはソースキューが作成されるまで、非アクティブのままでした。

JBoss EAP 8 では、ソースキューを作成してから、**bridge:add** コマンドを使用して JMS ブリッジを作成する必要があります。JMS ブリッジを作成してからソースキューを作成すると、**bridge:add** コマンドが失敗します。

message-activemq サブシステムへの新しいコネクタの追加

JBoss EAP 8.0 では、CLI を使用して **messaging-activemq** サブシステムの設定モデルに新しいコネクタを追加した場合、システムの他の部分がコネクタにアクセスできるように、サーバーを再起動ま

たはリロードする必要があります。JBoss EAP 7.4 では、コネクタが追加され、システムの他の部分によって参照されますが、サーバーを再起動またはリロードしないとコネクタを使用できません。

3.19. JAKARTA FACES の実装

MyFaces の Jakarta Faces 実装の変更

以前のリリースでは、Jakarta Faces 実装を別の実装に置き換えることができました。しかし、JBoss EAP 8.0 の **MyFaces** については、この機能が、Galleon ツールを使用したプロビジョニングを必要とする外部機能パックに移動しました。デフォルト以外の Mojarra バージョンを使用する場合は、手動設定が必要です。詳細は、[How to configure the Multi-JSF feature in EAP 8](#) を参照してください。

3.20. 高可用性

JGroup プロトコルスタックの更新

新しい "RED" プロトコルが JBoss EAP 8.0 の JGroup プロトコルスタックに追加されました。さらに、既存のプロトコルもアップグレードされました。

次の表に、プロトコルの更新を示します。

古いプロトコル	アップグレードされたプロトコル
FD SOCK	FD SOCK2
FD_ALL	FD_ALL3
VERIFY_SUSPECT	VERIFY_SUSPECT2
FRAG3	FRAG4

古いプロトコルスタックも JBoss EAP 8.0 で引き続き機能しますが、最適な結果を得るにはアップグレードされたスタックを使用してください。

3.21. JBOSS-EAP-INSTALLATION-MANAGER

jboss-eap-installation-manager を使用して、JBoss EAP 8.0 をインストールおよび更新できるようになりました。更新、復元、さまざまなチャネル管理タスクなどのサーバー管理操作を実行することもできます。

詳細は、[インストールガイド](#) を参照してください。

3.22. JBOSS-EAP-INSTALLATION-MANAGER の管理 CLI 統合

JBoss EAP 8.0 では、**installer** コマンドによる **jboss-eap-installation-manger** と管理 CLI の統合により、大幅な機能拡張が導入されました。この機能拡張により、スタンドアロンモードまたはマネージドドメインモードで、チャネル操作の更新、復元、管理などの幅広いサーバー管理操作をシームレスに行うできるようになります。

詳細は、[更新ガイド](#) を参照してください。

3.23. JBOSS-EAP-INSTALLATION-MANAGER の WEB コンソール統合

JBoss EAP 8.0 では、Web コンソールを使用して JBoss EAP インストール内のチャンネルを更新、復元、管理できるようになりました。ただし、**jboss-eap-installation-manager** を使用することを推奨します。

詳細は、[更新ガイド](#) を参照してください。

3.24. JBOSS EAP アプリケーションの移行

galleon/provisioning.xml 設定ファイルを使用して JBoss EAP 7.4 インストールを有効な S2I でプロビジョニングした場合、そのファイルを JBoss EAP 8 の有効な設定に変換するには、次の変更点に注意する必要があります。

- **galleon/provisioning.xml** 設定ファイルで、**eap-s2i** 機能パックの代わりに **org.jboss.eap:wildfly-ee-galleon-pack** および **org.jboss.eap:eap-cloud-galleon-pack** 機能パックを使用する必要があります。
- これらの機能パックを正常に使用するには、アプリケーション **pom.xml** で **eap-maven-plugin** を設定するか、S2I の環境変数を使用して、JBoss EAP 8 チャンネルの使用を有効にする必要もあります。

関連情報

- [Galleon プロビジョニングファイル](#)
- [従来の S2I プロビジョニング機能を使用して S2I ビルドを作成する](#)
- [Maven プラグインの設定属性](#)

第4章 サポート対象外の機能、非推奨機能、廃止された機能

4.1. サポート対象外の機能

以下の機能は、Red Hat ではサポートされていません。

ロギング

JBoss EAP 8.0 では、Apache Log4j バージョン 1 API はサポートされません。アプリケーションが **log4j.jar** および Log4j 設定をアプリケーションの一部としてパッケージ化していない場合は、それらを更新する必要があります。アプリケーションの移行または更新の詳細は、Red Hat ナレッジベースのソリューション [Migration: Apache Log4j version 1 is no longer provided in EAP 8](#) を参照してください。

Agroal サブシステム

JBoss EAP 8.0 では、Agroal サブシステムがサポートされなくなりました。

4.2. 非推奨の機能

このリリースでは、一部の機能が非推奨になりました。つまり、今後これらの機能が強化されることはなく、将来のリリースで廃止になる可能性があります。詳細は、[Deprecated in Red Hat JBoss Enterprise Application Platform \(EAP\) 8](#) を参照してください。

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 のサポートポリシーに関する詳細は、Red Hat カスタマーポータルに記載されている [Red Hat JBoss Middleware の製品更新とサポートポリシー](#) を参照してください。

以下の機能が非推奨になりました。

JBoss Tools

JBoss Tools は JBoss EAP 8.0 で非推奨になりました。

4.3. 削除された機能

JBoss EAP 8.0 では、以下の機能が削除されます。

Jolokia および Prometheus

このリリースでは **Jolokia** と **Prometheus** が削除されました。これらの機能は削除されており、Red Hat ではサポートされなくなります。JBoss EAP サーバーは、サーバーメトリクスエンドポイント **<server address>:<management port>/metrics** を介してメトリクスを公開します。

環境変数

Red Hat は JBoss EAP 8.0 で次の環境変数を削除しました。

- **GALLEON_PROVISION_DEFAULT_FAT_SERVER**
- **AB_JOLOKIA_AUTH_OPENSIFT**
- **AB_JOLOKIA_CONFIG**
- **AB_JOLOKIA_DISCOVERY_ENABLED**
- **AB_JOLOKIA_HOST**
- **AB_JOLOKIA_HTTPS**

- **AB_JOLOKIA_ID**
- **AB_JOLOKIA_OFF**
- **AB_JOLOKIA_OPTS**
- **AB_JOLOKIA_PASSWORD**
- **AB_JOLOKIA_PASSWORD_RANDOM**
- **AB_JOLOKIA_PORT**
- **AB_JOLOKIA_USER**
- **AB_PROMETHEUS_ENABLE**
- **AB_PROMETHEUS_JMX_EXPORTER_CONFIG**
- **AB_PROMETHEUS_JMX_EXPORTER_PORT**
- **JGROUPS_ENCRYPT_SECRET**

JDK 8

JDK 8 は Red Hat JBoss Enterprise Application Platform 8.0 から削除されました。JDK 11 または JDK 17 が必要になりました。

レガシーセキュリティーレルム

レガシーセキュリティーレルムは JBoss EAP 8.0 から削除されました。代わりに、**elytron** サブシステムで提供されるセキュリティーレルムを使用してください。

詳細は、[アイデンティティストアを使用したアプリケーションと管理インターフェイスの保護](#) および [複数のアイデンティティストアを使用したアプリケーションと管理インターフェイスの保護](#) ガイドを参照してください。

Picketbox

PicketBox は Red Hat JBoss Enterprise Application Platform 8.0 から削除されました。すべてのレガシーセキュリティー設定を、**elytron** サブシステムに移行する必要があります。セキュリティー設定を **elytron** サブシステムに移行する方法について、詳しくは [Elytron への移行](#) を参照してください。

PicketBox Vault

PicketBox Vault は JBoss EAP 8.0 から削除されました。代わりに、**elytron** サブシステムによって提供される認証情報ストアを使用して、機密性の高い文字列を保存します。

詳細は、[JBoss EAP における認証情報のセキュアなストレージガイドの Elytron のクレデンシャルとクレデンシャルストア](#) を参照してください。

PicketLink サブシステム

PicketLink サブシステムは JBoss EAP 8.0 から削除されました。PicketLink アイデンティティプロバイダーの代わりに Red Hat build of Keycloak を使用し、PicketLink サービスプロバイダーの代わりに Keycloak SAML アダプター機能によって提供される Galleon レイヤーを使用します。

詳細は、[JBoss EAP での Single Sign-On の使用ガイドの SAML によるアプリケーションの保護](#) を参照してください。

discovery-group と broadcast-group リソース

Red Hat JBoss Enterprise Application Platform 7.4 では、**discovery-group** と **broadcast-group** リソースが削除されました。これらのリソースは、JBoss EAP 8.0 でも削除されています。

さらに、Red Hat JBoss Enterprise Application Platform 7.4 は、**discovery-group** および **broadcast-group** リソースのすべてのインスタンスを **jgroups-discovery-group** および **socket-discovery-group** リソースに置き換えることで、Web コンソールへの影響を軽減しました。

JBoss EAP 7.3 では、**messaging** サブシステムで以下のリソースが非推奨になりました。

- **/subsystem=messaging-activemq/discovery-group=***
- **/subsystem=messaging-activemq/server=default/broadcast-group=***
- **/subsystem=messaging-activemq/server=default/discovery-group=***

JBoss EAP 7.3 は、これらの非推奨のリソースを **jgroups-discovery-group** および **socket-discovery-group** リソースに置き換えました。非推奨の各リソースには、各代替リソースの属性が含まれており、一方の属性は **null** に設定され、もう一方の属性は **0** より大きい値に設定されていました。これらの設定により、**discovery-group** と **broadcast-group** の両方がアクティブなままになりますが、それらのすべての機能が **jgroups-discovery-group** および **socket-discovery-group** リソースに割り当てられます。

クイックスタート

JBoss EAP 8.0 では、以下の古いまたは重複するクイックスタートが削除されました。

- **app-client**
- **bean-validation**
- **ejb-asynchronous**
- **ejb-in-ear**
- **ejb-in-war**
- **ejb-security**
- **ejb-security-jaas**
- **greeter**
- **helloworld-html5**
- **helloworld-mbean**
- **helloworld-mdb-propertysubstitution**
- **helloworld-rs**
- **helloworld-ssl**
- **inter-app**
- **jaxws-addressing**
- **jaxws-pojo**
- **jts-distributed-crash-rec**
- **kitchensink-angularjs**

- **kitchensink-ear**
- **kitchensink-jsp**
- **kitchensink-ml**
- **logging-tools**
- **managed-executor-service**
- **messaging-clustering**
- **payment-cdi-event**
- **resteasy-jaxrs-client**
- **spring-greeter**
- **spring-kitchensink-basic**
- **spring-kitchensink-springmvctest**
- **tasks-rs**
- **websocket-client**
- **xml-jaxp**

Red Hat build of Keycloak クライアントアダプター

Red Hat build of Keycloak クライアントアダプターは、Red Hat JBoss Enterprise Application Platform 8.0 では提供されません。新しい **elytron-oidc-client** サブシステムを使用し、OpenID Connect (OIDC) を使用して JBoss EAP にデプロイされたアプリケーションを保護してください。

Red Hat Enterprise Linux 上の Java サービス

Red Hat Enterprise Linux (RHEL) 上で実行される Java サービス (JSVC) は、JBoss EAP 8.0 から削除されました。

BOM

次の BOM が削除されました。

- **JBoss Jakarta EE 8 Specification APIs** BOM が削除されました。Maven プロジェクトでは **JBoss EAP EE** BOM を使用してください。
- **EAP Runtime Artifacts** BOM が削除されました。Maven プロジェクトでは **JBoss EAP EE** BOM を使用してください。
- **JBoss EJB client legacy** BOM が削除されました。

詳細は、[移行ガイドの JBoss EAP アプリケーションの Maven プロジェクトを JBoss EAP 8.0 に移行する](#) を参照してください。

connector 属性

JBoss EAP 7.4 で、modcluster サブシステムによってプロキシ要素の **connector** 属性が非推奨となり、混乱を避けるためにこの属性が **listener** 属性に置き換えられました。JBoss EAP 7.4 の管理スキーマは、**listener** 属性を使用しますが、**connector** 属性の設定も許可します。JBoss EAP 8.0 では、非推奨の **connector** 属性が削除されたため、代わりに **listener** 属性を使用する必要があります。詳細は、[Deprecated in Red Hat JBoss Enterprise Application Platform \(EAP\) 7](#) を参照してください。

iiop-openjdk サブシステムの変更

JBoss EAP 8.0 では、レガシーセキュリティーサブシステムが削除され、**elytron** サブシステムに置き換えられました。**elytron** サブシステムは、Object Resource Broker (ORB) のセキュリティーインターセプターとしてインストールできます。

JBoss EAP 7 との相互運用性を維持するために (たとえば JBoss EAP をマネージドドメインとして実行し、JBoss EAP 8.0 ホストコントローラーで JBoss EAP 7 セカンダリーホストコントローラーを管理するときのために)、レガシーセキュリティーインターセプターを設定する機能が保持されています。

ただし、JBoss EAP をスタンドアロンサーバーとして実行する場合、**iiop-openjdk/security** 属性の値 **client** および **identity** の設定は **サポートされません**。

Hibernate Search 5 API

Hibernate Search 5 API は、JBoss EAP 7.4 で非推奨となり、JBoss EAP 8.0 で削除され、Hibernate Search 6 API に置き換えられました。



注記

Hibernate Search 6 API には、Hibernate Search 5 API との **下位互換性はありません**。運用性を維持するには、アプリケーションを Hibernate Search 6 に移行する必要があります。アプリケーションを移行するには、[Hibernate Search 6.0 の移行ガイド](#) を参照してください。

削除された API のリストを確認するには、[Hibernate Search 5 APIs Deprecated in JBoss EAP 7.4 and removed in EAP 8.0](#) を参照してください。

Apache Log4j バージョン 1

JBoss EAP 8.0 では、Apache Log4j バージョン 1 API はサポートされません。**log4j.jar** および Log4j 設定がアプリケーションの一部としてパッケージ化されていない場合は、これらのパッケージを更新する必要があります。アプリケーションの移行または更新の詳細は、Red Hat ナレッジベースのソリューション [Migration: Apache Log4j version 1 is no longer provided in EAP 8](#) を参照してください。

詳細は、[Apache Log4j バージョン 1 API の削除](#) を参照してください。

Apache Xerces と Apache Xalan

JAXP バージョン 1.5 を実装する Apache Xerces および Apache Xalan JBoss モジュールは、JBoss EAP 8.0 から削除されました。代わりに、JAXP バージョン 1.6 を実装する JDK の **java.xml** JPMS モジュールによって提供されるデフォルトの JAXP 実装を使用してください。詳細は、[Use JAXP implementation provided by the JDK in JBoss EAP](#) を参照してください。

第5章 解決した問題

お客様のケースから発信され、このリリースで解決した問題のリストを表示するには、[Resolved Issues for Red Hat JBoss Enterprise Application Platform 8.0](#) を参照してください。

第6章 修正された CVE

JBoss EAP 8.0 には、以下のセキュリティー関連の問題修正が含まれています。

- [JBEAP-25077:\(8.0.z\) Upgrade netty from 4.1.92.Final to 4.1.94.Final \(resolves CVE-2023-34462\)](#)
- [JBEAP-25748:\(8.0.z\) JBWS-4389 - Wrong assumption about the Identity's password are all clearpassword](#)
- [JBEAP-25356:\(8.0.0\) Upgrade Guava from 31.1.0.jre-redhat-00001 to 32.1.1.jre-redhat-00001](#)
- [JBEAP-25646:\(8.0.z\) Upgrade FasterXML Jackson from 2.14.2.redhat-00001 to 2.15.2](#)
- [JBEAP-25943: Upgrade santuario to 3.0.3](#)
- [JBEAP-24435: MP OpenAPI - Loading static files bigger than 3MB fails since SmallRye OpenAPI 3.0.1 uses new SnakeYaml that sets a constraint](#)

第7章 既知の問題

このリリースの既知の問題のリストを確認するには、[Known Issues for Red Hat JBoss Enterprise Application Platform 8.0](#) を参照してください。

7.1. INFINISPAN

`/subsystem=distributable-web/infinispan-session-management=*:add` 操作をデフォルトの非 HA サーバー設定で実行すると失敗する場合があります

問題 - JBEAP-24997

`/subsystem=distributable-web/infinispan-session-management=*:add` 操作は、`affinity=primary-owner` 子リソースを自動的に追加します。これには `routing=infinispan` リソースが必要です。この操作は失敗する可能性があります。必要な `routing=infinispan` リソースが、デフォルトの非 HA サーバー設定で定義されていないためです。

回避策

この無効な中間状態を回避するには、`infinispan-session-management:add` 操作と `affinity=local:add` 操作の両方を 1つのバッチ内で実行します。

以下に例を示します。

```
batch
/subsystem=distributable-web/infinispan-session-management=ism-0:add(cache-
container=web,granularity=SESSION)
/subsystem=distributable-web/infinispan-session-management=ism-0/affinity=local:add()
run-batch -v
```

HotRod が Infinispan への外部化のための分散セッションを作成できない

問題 - JBEAP-26062

OpenShift Container Platform 上の Red Hat JBoss Enterprise Application Platform 8.0 と Red Hat Data Grid が関与する相互運用性テストで、Infinispan リモートキャッシュへの書き込みによって内部サーバーエラーが発生する問題が明らかになりました。`remote-cache-container` がデフォルトのマーシャラーである JBoss Marshalling を使用するように設定されている場合、`byte[]` インスタンスしかサポートされていないため、キャッシュ書き込みが原因で HotRod からエラーが出力されません。

エラーメッセージの例:

```
Caused by: java.lang.IllegalArgumentException: Only byte[] instances are supported currently!
at org.infinispan.client.hotrod@14.0.17.Final-redhat-00002/org.infinispan.client.hotrod.marshall.BytesOnlyMarshaller.checkByteArray(BytesOnlyMarshaller.java:27)
```

回避策

ProtoStream マーシャラー `marshaller=PROTOSTREAM` を使用するように `remote-cache-container` を設定します。

設定例:

```
/subsystem=infinispan/remote-cache-container=
<RHDG_REMOTE_CACHE_CONTAINER_RESOURCE_NAME>:write-
attribute(name=marshaller,value=PROTOSTREAM)
```

7.2. データソース設定

MySQL の接続の回復性がサポートされていない

問題 - JBEAP-25585

Red Hat JBoss Enterprise Application Platform 8.0 は、MySQL JDBC ドライバーバージョン 10.2.0 以降の接続の回復性をサポートしていません。接続の回復性が有効になっていると、このドライバーが回復マネージャーにとって想定外の状態になります。このドライバーの接続の回復性は、デフォルトで有効になっており、ユーザーが手動で無効にする必要があります。

回避策

ConnectRetryCount パラメーターは、接続障害が発生した場合の再接続の試行回数を制御します。このパラメーターはデフォルトで **1** に設定されています。これにより接続の回復性が有効になります。

接続の回復性を無効にするには、**ConnectRetryCount** パラメーターを **1** から **0** に変更します。接続プロパティは、サーバー設定ファイル **standalone.xml** または **domain.xml** のデータソース設定セクションで設定できます。データソース設定の設定方法の詳細は、Red Hat カスタマーポータル [How to configure datasource settings in EAP for OpenShift](#) および [How to specify connection properties in the Datasource Configuration for JBoss EAP](#) を参照してください。

7.3. サーバー管理

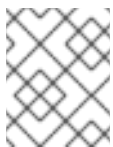
liveness プロブ :9990/health/live がデプロイメントエラー発生時に Pod を再起動しない

問題 - JBEAP-24257

JBoss EAP 7.4 では、コンテナの再起動の原因となるデプロイメントエラーが発生した場合、Python liveness プロブが "not alive" と報告します。

JBoss EAP 8.0 では、liveness プロブ **:9990/health/live** がサーバー管理モデルを使用して準備状況を判断します。サーバー状態が実行中であり、ブートエラーもデプロイメントエラーもない場合、liveness チェックは、サーバープロセスが実行中の間は **UP** と報告します。

したがって、デプロイメントエラーにより、実行中であっても "not ready" 状態の Pod が発生する可能性があります。これは、デプロイ中に断続的なエラーが発生するアプリケーションにのみ影響します。デプロイ中にこのようなエラーが常に発生する場合、コンテナが準備完了状態にならず、Pod が **CrashLoopBackoff** 状態になります。



注記

:9990/health/live は、Helm チャートと JBoss EAP Operator によって使用されるデフォルトの liveness プロブです。

回避策

デプロイメントエラーが発生し、実行中の Pod が "not ready" と報告される場合は、サーバーのブートプロセスを調べ、エラーの原因となっているデプロイの問題を解決し、サーバーが正しくデプロイされることを確認します。

デプロイメントエラーを修正できない場合は、**/ready** HTTP エンドポイントを使用するように Startup Probe を変更して、ブートエラー発生時に Pod が再起動するようにします。たとえば、

Helm を使用して JBoss EAP アプリケーションをデプロイする場合は、**deploy.livenessProbe** フィールドを更新して liveness プロブを設定します。

```
deploy:
  livenessProbe:
    httpGet:
      path: /health/ready
```

7.4. メッセージングフレームワーク

org.apache.activemq.artemis モジュールの非推奨および警告メッセージ

問題 - [JBEAP-26188](#)

org.apache.activemq.artemis モジュールは、JBoss EAP 8.0 で非推奨になりました。MANIFEST.MF または **jboss-deployment-structure.xml** 設定ファイルにこのモジュールの依存関係を持つアプリケーションをデプロイすると、警告メッセージがトリガーされます。詳細は、[Deprecated in Red Hat JBoss Enterprise Application Platform \(EAP\) 8](#) を参照してください。

回避策

JBoss EAP 8.0 Update 1 では、設定ファイル内の **org.apache.activemq.artemis** モジュールを **org.apache.activemq.artemis.client** パブリックモジュールに置き換えることで、このような警告メッセージのロギングを防ぐことができます。詳細は、[org.jboss.as.dependency.deprecated ... is using a deprecated module \("org.apache.activemq.artemis"\) in EAP 8](#) を参照してください。

7.5. IBM MQ リソースアダプター

IBM MQ リソースアダプターの制限と既知の問題

IBM MQ リソースアダプターはサポートされていますが、いくつかの制限があります。詳細は、[IBM MQ リソースアダプターのデプロイ](#) を参照してください。

改訂日時: 2024-02-08