



# Red Hat JBoss Web Server 5.7

## インストールガイド

Red Hat JBoss Web Server 6.0 のインストールおよび設定



# Red Hat JBoss Web Server 5.7 インストールガイド

---

Red Hat JBoss Web Server 6.0 のインストールおよび設定

## 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<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

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

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> 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<sup>®</sup> 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

サポートされているオペレーティングシステムで Red Hat JBoss Web Server のインストール、アップグレード、および基本設定を実行します。

## 目次

RED HAT JBOSS WEB SERVER ドキュメントへのフィードバック .....	4
多様性を受け入れるオープンソースの強化 .....	5
<b>第1章 RED HAT JBOSS WEB SERVER インストールの概要 .....</b>	<b>6</b>
1.1. JBOSS WEB SERVER のコンポーネント	6
1.2. RED HAT が提供する APACHE TOMCAT ディストリビューション間の違い	6
1.3. JBOSS WEB SERVER のオペレーティングシステムと設定	9
1.4. JBOSS WEB SERVER のインストール方法	9
1.5. JBOSS WEB SERVER コンポーネントドキュメントバンドル	9
<b>第2章 アーカイブファイルからの RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER のインストール .....</b>	<b>11</b>
2.1. 前提条件	11
2.2. RHEL での JBOSS WEB SERVER アーカイブファイルのダウンロードと解凍	13
2.3. アーカイブファイルからインストールした場合に SYSTEMD を使用して JBOSS WEB SERVER を管理する	14
2.4. コマンドラインからアーカイブのインストールを管理するための JBOSS WEB SERVER 設定	15
2.5. アーカイブファイルからインストールした場合のコマンドラインからの JBOSS WEB SERVER の起動	17
2.6. アーカイブファイルからインストールした場合にコマンドラインから JBOSS WEB SERVER を停止する	18
2.7. JBOSS WEB SERVER の SELINUX ポリシー	18
2.8. TOMCAT ユーザーおよびグループの UID および GID の変更	20
<b>第3章 RPM パッケージからの RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER のインストール ...</b>	<b>23</b>
3.1. 前提条件	23
3.2. RED HAT ENTERPRISE LINUX へのサブスクリプションの割り当て	25
3.3. YUM を使用した RPM パッケージからの JBOSS WEB SERVER のインストール	26
3.4. RPM からインストールした場合の JBOSS WEB SERVER の起動	27
3.5. RPM からインストールした場合の JBOSS WEB SERVER の停止	28
3.6. システムの起動時に JBOSS WEB SERVER サービスを開始するように設定する	28
3.7. JBOSS WEB SERVER の SELINUX ポリシー	29
<b>第4章 MICROSOFT WINDOWS への JBOSS WEB SERVER のインストール .....</b>	<b>31</b>
4.1. MICROSOFT WINDOWS への JDK のインストール	31
4.2. MICROSOFT WINDOWS での JBOSS WEB SERVER のダウンロードと抽出	31
4.3. MICROSOFT WINDOWS での JBOSS WEB SERVER の設定	32
4.4. MICROSOFT WINDOWS での JBOSS WEB SERVER の起動	34
4.5. MICROSOFT WINDOWS での JBOSS WEB SERVER の停止	35
<b>第5章 JBOSS WEB SERVER 用の HIBERNATE の設定 .....</b>	<b>36</b>
5.1. HIBERNATE ORM のインストール	36
5.2. JDBC 接続プールの設定	36
5.3. HIBERNATE 接続プロパティの設定	37
5.4. JDBC データソースの追加	38
<b>第6章 RED HAT JBOSS WEB SERVER の HTTP/2 の有効化 .....</b>	<b>39</b>
6.1. 前提条件	39
6.2. コネクタの HTTP/2 を有効にする	39
6.3. JBOSS WEB SERVER ログを表示して、HTTP/2 が有効になっていることを確認する	41
6.4. CURL コマンドを使用して HTTP/2 が有効になっていることを確認する	41
6.5. 関連情報 (または次の手順)	42
<b>第7章 RED HAT JBOSS WEB SERVER でのパスワード VAULT の使用 .....</b>	<b>43</b>
7.1. アーカイブファイルからのパスワード VAULT のインストール	43
7.2. YUM パッケージマネージャーを使用して RHEL にパスワード VAULT をインストールする	43

---

7.3. JBOSS WEB SERVER でパスワード VAULT を有効にする	43
7.4. JBOSS WEB SERVER での JAVA キーストアの作成	44
7.5. APACHE TOMCAT のパスワード VAULT の初期化	45
7.6. パスワード VAULT を使用するための TOMCAT の設定	47
7.7. 外部パスワード VAULT の設定	48
7.8. パスワード VAULT への SENSITIVE 文字列の格納	48
7.9. TOMCAT 設定で保存された機密文字列の使用	49
<b>第8章 SSI フィルターの設定</b> .....	<b>50</b>
<b>第9章 RED HAT JBOSS WEB SERVER の FIPS の設定</b> .....	<b>51</b>
9.1. FIPS の概要	51
9.2. RHEL 8 での JBOSS WEB SERVER の FIPS の設定	51
<b>付録A JAVA IPV4 および IPV6 プロパティ</b> .....	<b>53</b>
A.1. JAVA IPV4 および IPV6 プロパティの概要	53
A.2. JAVA IPV4 および IPV6 プロパティの TOMCAT へのエクスポート	53
A.3. TOMCAT バインディングの設定	53



## RED HAT JBOSS WEB SERVER ドキュメントへのフィードバック

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。詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

## 第1章 RED HAT JBOSS WEB SERVER インストールの概要

Red Hat JBoss Web Server は、Java Web アプリケーションをホストする完全に統合および認定されたコンポーネントセットです。Red Hat JBoss Web Server は、Apache Tomcat サーブレットコンテナと Tomcat ネイティブライブラリーの完全にサポートされた実装を提供します。



### 注記

Java アプリケーションのクラスタリングまたはセッション複製のサポートが必要な場合は、Red Hat JBoss Enterprise Application Platform (JBoss EAP) を使用してください。

### 1.1. JBOSS WEB SERVER のコンポーネント

JBoss Web Server には、Apache Tomcat サーブレットコンテナ、Tomcat ネイティブライブラリー、Tomcat vault、**mod\_cluster** ライブラリー、Apache Portable Runtime (APR)、OpenSSL などのコンポーネントが含まれています。

#### Apache Tomcat

Apache Tomcat は、Java Servlet Specification に準拠したサーブレットコンテナです。JBoss Web Server には Apache Tomcat 9 が含まれています。

#### Tomcat native library

Tomcat ネイティブライブラリーは、Tomcat のスケーラビリティ、パフォーマンス、およびネイティブサーバー技術との統合を向上させます。

#### Tomcat vault

Tomcat vault は、JBoss Web Server で使用されるパスワードやその他の機密情報を安全に保管するために使用される JBoss Web Server 用の拡張機能です。

#### Mod\_cluster

**mod\_cluster** ライブラリーは、Apache Tomcat と Apache HTTP Server の **mod\_proxy\_cluster** モジュール間の通信を可能にします。**mod\_cluster** ライブラリーにより、Apache HTTP Server を JBoss Web Server のロードバランサーとして使用することができます。**mod\_cluster** の設定の詳細、または **mod\_jk** や **mod\_proxy** などの代替ロードバランサーのインストールと設定の詳細は、[HTTP コネクタとロードバランシングガイド](#) を参照してください。

#### Apache ポート可能なランタイム

Apache ポータブルランタイム (APR): ネイティブサーバー技術との優れたスケーラビリティ、パフォーマンス、および改善された統合を提供するランタイム。APR は、Apache HTTP Server 2.x の中核となる大変ポータブルなライブラリーです。これにより、sendfile、epoll、OpenSSL などの高度な IO 機能、乱数生成やシステムステータスなどのオペレーティングシステムレベルの機能、共有メモリー、NT パイプ、UNIX ソケットなどのネイティブプロセス処理機能にアクセスできます。

#### OpenSSL

OpenSSL は、Secure Sockets Layer (SSL) および Transport Layer Security (TLS) プロトコルを実装するソフトウェアライブラリーです。OpenSSL には、基本的な暗号化ライブラリーが含まれています。

Red Hat JBoss Web Server がサポートするコンポーネントの完全なリストについては、[JBoss Web Server Component Details](#) を参照してください。

### 1.2. RED HAT が提供する APACHE TOMCAT ディストリビューション間の違い

Red Hat JBoss Web Server と Red Hat Enterprise Linux (RHEL) は両方とも、Apache Tomcat の個別の

ディストリビューションを提供します。ただし、JBoss Web Server には、統合および認定された追加のコンポーネントと機能のセットが含まれているため、Apache Tomcat の RHEL ディストリビューションと比較して明確な利点があります。JBoss Web Server は、より頻繁にソフトウェアとセキュリティの更新を提供します。



### 注記

RHEL は、RHEL 7、RHEL 8.8、および RHEL 9.2 以降でのみ Apache Tomcat のディストリビューションを提供します。

RHEL 8.0 ~ 8.7 および RHEL 9.0 ~ 9.1 については、RHEL プラットフォームサブスクリプションでは Apache Tomcat のディストリビューションが提供されていません。これらのオペレーティングシステムバージョンでは、JBoss Web Server は Red Hat が提供する唯一の Apache Tomcat ディストリビューションであり、Middleware Runtimes サブスクリプションの一部として利用できます。

## Apache Tomcat バージョン

JBoss Web Server および RHEL で使用できる Apache Tomcat ディストリビューションの次のバージョン情報を考慮してください。

- RHEL 7 **t omcat** パッケージは、Apache Tomcat 7 のコミュニティバージョンに基づいています。
- RHEL 8.8 および RHEL 9.x **tomcat** パッケージは、Apache Tomcat 9 のコミュニティバージョンに基づいています。
- JBoss Web Server 3.1 は、Apache Tomcat 7 および Apache Tomcat 8 のディストリビューションと、追加のコンポーネントおよび機能の統合および認定済みのセットを提供します。ただし、Red Hat は JBoss Web Server 3.1 を完全にサポートまたは保守していません。JBoss Web Server 3.1 は現在、延長ライフサイクルサポート (ELS) フェーズ 2 にあり、サポート終了日は 2028 年 12 月に予定されています。
- JBoss Web Server 5.x は、Red Hat が完全にテストおよびサポートする Apache Tomcat 9 のディストリビューションを、統合および認定された追加のコンポーネントおよび機能のセットとともに提供します。



### 注記

Red Hat は、Apache Tomcat のコミュニティリリースのサポートを提供しません。

詳細は、[Red Hat でサポートされる Apache Tomcat のバージョン](#) を参照してください。

## JBoss Web Server と Apache Tomcat の RHEL ディストリビューションの違い

JBoss Web Server と Apache Tomcat の RHEL ディストリビューションの次の違いを考慮してください。

- JBoss Web Server は、アーカイブファイルまたは RPM パッケージから RHEL バージョン 7、8、および 9 にインストールできます。Apache Tomcat の RHEL ディストリビューションは、RHEL 7、RHEL 8.8、および RHEL 9.2 以降の RPM パッケージからのみインストールできません。
- アーカイブファイルから、サポートされている Windows Server プラットフォームに JBoss Web Server をインストールすることもできます。
- Apache Tomcat の RHEL ディストリビューションは、RHEL システム上で Apache Tomcat イ

ンスタンスをデプロイおよび実行するための管理者サポートを提供します。JBoss Web Server は、安全で安定した環境で Apache HTTP Server プロキシからのクライアントリクエストに サービスを提供できるバックエンド Web アプリケーションと大規模な Web サイトの作成とデプロイメントを開発者に提供します。

- RHEL では、Apache Tomcat の標準ディストリビューションのみが提供され、コミュニティバージョンに基づいた頻度の低いソフトウェア更新が提供されます。JBoss Web Server は、完全にテストされサポートされている Apache Tomcat のディストリビューションを提供します。これには、以下の統合および認定された追加機能のセットが含まれています。
  - **mod\_proxy**、**mod\_jk**、または **mod\_proxy\_cluster** コネクタを使用して、Web クライアントリクエストをバックエンド Web アプリケーションに転送およびロードバランシングするための、完全にテストおよび認定された Apache HTTP サーバーとの統合
  - Apache Tomcat のスケーラビリティ、パフォーマンス、およびネイティブサーバーテクノロジーとの統合を向上させるための Tomcat ネイティブライブラリー
  - パスワードやその他の機密文字列をマスクし、暗号化された Java キーストアに機密情報を安全に保存するための Tomcat Vault 拡張機能
  - Apache HTTP サーバーの **mod\_proxy\_cluster** モジュールとバックエンド JBoss Web Server のワーカーノード間の通信と Web トラフィックのインテリジェントな負荷分散を可能にする **Mod\_cluster** ライブラリー
  - Apache Portable Runtime (APR) ライブラリーにより、優れたスケーラビリティ、パフォーマンス、およびネイティブサーバーテクノロジーとの統合の向上を実現
  - Federal Information Processing Standards (FIPS) コンプライアンス
  - Red Hat OpenShift 環境での JBoss Web Server のサポート
  - OpenShift コンテナイメージを管理し、Red Hat OpenShift 環境で Web サーバーアプリケーションのインスタンスを作成、設定、管理、シームレスにアップグレードするための JBoss Web Server Operator
  - Red Hat Ansible 認定コンテンツコレクションを使用した JBoss Web Server の自動インストール
- JBoss Web Server は、**jws-5.X.x-maven-repository.zip** ファイルで、一連の Maven レポジトリアーティファクトを提供しています。このファイルは、Red Hat カスタマーポータルからダウンロードできます。これらのアーティファクトは、アプリケーションデプロイメントプロジェクトの Web アプリケーションアーカイブ (WAR) ファイルで使用できます。Apache Tomcat の RHEL ディストリビューションは、Maven レポジトリアーティファクトのセットを提供しません。
- JBoss Web Server では、**jws-5.X.x-maven-repository.zip** ファイルに組み込み Tomcat のライブラリーが含まれています。これにより、完全にサポートされている Apache Tomcat バージョンで組み込み Tomcat を使用して Web アプリケーションを構築できます。

### JBoss Web Server と RHEL ドキュメントセットの違い

JBoss Web Server ドキュメントセットは、**tomcat** パッケージの RHEL ドキュメントよりも幅広く、より包括的です。

- JBoss Web Server には、**Red Hat JBoss Web Server 6.0.x Documentation** アーカイブファイルが含まれます。これには、Apache Tomcat 10.1 と Tomcat Vault の API ドキュメントが含まれています。このアーカイブファイルは [Red Hat Customer Portal](#) からダウンロードできます。

- JBoss Web Server の [製品ドキュメントページ](#) では、次のすべてのタイプのユースケースに関する情報が提供されます。
  - サポートされているオペレーティングシステム上で、アーカイブファイルまたは RPM パッケージから JBoss Web Server の標準インストールを実行します。
  - Apache HTTP Server コネクタおよび **mod\_jk** や **mod\_proxy\_cluster** などのロードバランサーとともに使用するための JBoss Web Server を設定します。
  - Red Hat Ansible 認定コンテンツコレクションを使用して、JBoss Web Server の自動インストールを有効にします。
  - Red Hat OpenShift 環境で JBoss Web Server を使用します。
  - JBoss Web Server Operator for OpenShift のインストールと使用。
  - Hibernate オブジェクトリレーショナルマッピング (ORM) フレームワーク、HTTP/2 プロトコル、Tomcat Vault、FIPS 準拠などの機能をサポートするように JBoss Web Server を設定します。

### 1.3. JBOSS WEB SERVER のオペレーティングシステムと設定

Red Hat JBoss Web Server は、異なるバージョンの Red Hat Enterprise Linux および Microsoft Windows オペレーティングシステムをサポートします。

#### 関連情報

- [JBoss Web Server 5 でサポートされる設定](#)

### 1.4. JBOSS WEB SERVER のインストール方法

各プラットフォームで利用可能なアーカイブインストールファイルを使用して、サポートされている Red Hat Enterprise Linux および Microsoft Windows システムに Red Hat JBoss Web Server をインストールできます。RPM パッケージを使用して、サポートされる Red Hat Enterprise Linux システムに JBoss Web Server をインストールすることもできます。

以下のコンポーネントはアーカイブインストールファイルに含まれます。これらのコンポーネントは、JBoss Web Server インストールのコア部分です。

- **jws-5.7.0-application-server.zip**
  - Apache Tomcat 9
  - **mod\_cluster**
  - Tomcat vault
- **jws-5.7.0-application-server-<platform>-<architecture>.zip**
  - プラットフォーム固有のユーティリティ

### 1.5. JBOSS WEB SERVER コンポーネントドキュメントバンドル

JBoss Web Server には、各コンポーネントの元のベンダードキュメントが含まれる追加のドキュメントバンドルが含まれています。このドキュメントバンドル **jws-docs-5.7.0.zip** は、[Red Hat カスタマーポータル](#) からダウンロードできます。

ドキュメントバンドルには、次のコンポーネントの追加ドキュメントが含まれています。

- Apache Tomcat
- Tomcat native library
- Tomcat vault

## 第2章 アーカイブファイルからの RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER のインストール

アーカイブファイルまたは RPM パッケージから Red Hat Enterprise Linux に JBoss Web Server をインストールできます。アーカイブファイルから JBoss Web Server をインストールする場合は、[Red Hat Customer Portal](#) から JBoss Web Server アーカイブファイルをダウンロードして展開できます。

JBoss Web Server をアーカイブファイルからインストールすると、さまざまな方法で製品を管理できます。たとえば、システム起動時にシステムデーモンを使用したり、コマンドラインから JBoss Web Server を管理したりできます。

### 2.1. 前提条件

- YUM パッケージマネージャーを使用するか、圧縮されたアーカイブから、サポートされている Java Development Kit (JDK) をインストールしている。
- 使用中のシステムが Red Hat Enterprise Linux パッケージ要件に準拠している。

#### 2.1.1. YUM パッケージマネージャーを使用した JDK のインストール

YUM パッケージマネージャーを使用して、Java Development Kit (JDK) をインストールできます。サポートされている JDK の完全なリストについては、[JBoss Web Server operating systems and configurations](#) を参照してください。

#### 手順

1. Red Hat Enterprise Linux システムを適切なチャンネルにサブスクライブします。

- **OpenJDK:**
  - rhel-7-server-rpms
  - rhel-8-server-rpms
  - rhel-9-server-rpms
- **IBM:**
  - rhel-7-server-supplementary-rpms
  - rhel-8-server-supplementary-rpms
  - rhel-9-server-supplementary-rpms



#### 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

2. root ユーザーとしてコマンドを実行し、1.8 JDK をインストールします。

```
# yum install java-1.8.0-<VENDOR>-devel
```

<VENDOR> を **ibm** または **openjdk** に置き換えます。

3. root ユーザーとして以下のコマンドを実行し、正しい JDK が使用されていることを確認します。

```
# alternatives --config java
```

```
# alternatives --config javac
```

これらのコマンドは、利用可能な JDK バージョンと、プラス (+) 記号の付いた選択したバージョンのリストを返します。選択した JDK が必要な JDK ではない場合は、シェルプロンプトで指示されたように、必要な JDK に変更します。



### 重要

**java** コマンドおよび **javac** コマンドを使用するすべてのソフトウェアは、**alternatives** によって設定された JDK を使用します。Java の代替の変更は、他のソフトウェアの実行に影響を及ぼす可能性があります。

## 2.1.2. 圧縮アーカイブからの JDK のインストール

**.zip** または **.tar** ファイルなどの圧縮アーカイブから Java Development Kit (JDK) をインストールできます。サポートされている JDK の完全なリストについては、[JBoss Web Server operating systems and configurations](#) を参照してください。

### 手順

1. JDK がベンダーの Web サイト (Oracle または OpenJDK) からダウンロードされている場合は、ベンダーが提供するインストール手順を使用して、**JAVA\_HOME** 環境変数を設定します。
2. JDK が圧縮されたアーカイブからインストールされた場合は、Tomcat の **JAVA\_HOME** 環境変数を設定します。
  - a. Tomcat の **bin** ディレクトリー (**JWS\_HOME/tomcat/bin**) で、**setenv.sh** という名前のファイルを作成します。
  - b. **setenv.sh** ファイルで、**JAVA\_HOME** パス定義を入力します。以下に例を示します。

```
$ cat JWS_HOME/tomcat/bin/setenv.sh
export JAVA_HOME=/usr/lib/jvm/jre-1.8.0-openjdk.x86_64
```

## 2.1.3. Red Hat Enterprise Linux パッケージの要件

Red Hat Enterprise Linux に JBoss Web Server をインストールする前に、システムが以下のパッケージ要件に準拠していることを確認する必要があります。



### 注記

パッケージの要件は、使用している Red Hat Enterprise Linux のバージョンによって異なります。

- Red Hat Enterprise Linux バージョン 8 または 9 で、OpenSSL または Apache Portable Runtime (APR) を使用する場合は、Red Hat Enterprise Linux が提供する **openssl** および **apr** パッケージをインストールする必要があります。



- **openssl** パッケージをインストールするには、root ユーザーとして次のコマンドを入力します。

```
# yum install openssl
```

- **apr** パッケージをインストールするには、root ユーザーとして次のコマンドを入力します。

```
# yum install apr
```

- **tomcat-native** パッケージをインストールする前に、**tomcatjss** パッケージを削除する必要があります。**tomcatjss** パッケージは、OpenSSL セキュリティーモデルではなく、基盤となる Network Security Services (NSS) セキュリティーモデルを使用します。**tomcatjss** パッケージを削除するには、root ユーザーとして次のコマンドを入力します。

```
# yum remove tomcatjss
```

### 注記

- Red Hat Enterprise Linux 7 では、JBoss Web Server は、Red Hat JBoss Core Services で提供される **openssl** および **apr** パッケージを使用します。
- Red Hat Enterprise Linux バージョン 8 および 9 では、JBoss Web Server は **openssl** および **apr** パッケージを提供しません。JBoss Web Server は、Red Hat Enterprise Linux が提供する openssl および apr パッケージを使用します。Red Hat Enterprise Linux バージョン 8 または 9 で OpenSSL または APR を使用する場合は、このセクションで前述したように、オペレーティングシステムから openssl および apr パッケージをインストールする必要があります。

## 2.2. RHEL での JBOSS WEB SERVER アーカイブファイルのダウンロードと解凍

JBoss Web Server アーカイブファイルは、[Red Hat Customer Portal](#) からダウンロードできます。

### 前提条件

- [YUM パッケージマネージャーを使用する](#) か、[圧縮されたアーカイブから](#)、サポートされている Java Development Kit (JDK) をインストールしている。
- 使用中のシステムが [Red Hat Enterprise Linux パッケージ要件](#) に準拠している。

### 手順

1. ブラウザーを開き、[Red Hat カスタマーポータル](#) にログインします。
2. **Downloads** をクリックします。
3. **製品のダウンロード** リストの **Red Hat JBoss Web Server** をクリックします。
4. **Version** ドロップダウンメニューから正しい JBoss Web Server バージョンを選択します。
5. 以下の各ファイルの **Download** をクリックし、システムのプラットフォームおよびアーキテクチャーが正しく選択されていることを確認します。

- The Red Hat JBoss Web Server 5.5 Application Server (**jws-5.5.0-application-server.zip**).
  - Red Hat JBoss Web Server 5.5 Native Components for RHEL (**jws-5.5.0-application-server-`<platform>`-`<architecture>`.zip**)
6. ダウンロードしたアーカイブファイルをインストールディレクトリーにデプロイメントします。  
以下に例を示します。

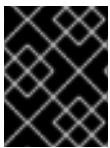
```
# unzip jws-5.5.0-application-server.zip -d /opt/
# unzip -o jws-5.5.0-application-server-<platform>-<architecture>.zip -d /opt/
```

アーカイブを解凍すると、JBoss Web Server の最上位ディレクトリーが作成されます。このドキュメントでは、JBoss Web Server の最上位ディレクトリーを **JWS\_HOME** と呼んでいます。

## 2.3. アーカイブファイルからインストールした場合に **SYSTEMD** を使用して **JBoss WEB SERVER** を管理する

Red Hat Enterprise Linux のアーカイブファイルから JBoss Web Server をインストールする場合、システムデーモンを使用して管理タスクを実行できます。JBoss Web Server をシステムデーモンで使用すると、システムの起動時に JBoss Web Server サービスを起動する方法を利用できます。システムデーモンは、start、stop、および status チェック機能も提供します。

Red Hat Enterprise Linux バージョン 8 および 9 では、デフォルトのシステムデーモンは **systemd** です。



### 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

### 前提条件

- [JBoss Web Server](#) をアーカイブファイルからインストールしました。

### 手順

1. 実行中のシステムデーモンを特定するには、次のコマンドを入力します。

```
$ ps -p 1 -o comm=
```

**systemd** が実行されている場合、次の出力が表示されます。

```
systemd
```

2. **systemd** 用に JBoss Web Server をセットアップするには、root ユーザーとして **.postinstall.systemd** スクリプトを実行します。

```
# cd JWS_HOME/tomcat
# sh .postinstall.systemd
```

3. **systemd** を使用して JBoss Web Server を制御するには、root ユーザーとして次のいずれかの手順を実行できます。

- **systemd** を使用してシステムの起動時に JBoss Web Server サービスを起動できるようにするには:

```
# systemctl enable jws5-tomcat.service
```

- **systemd** を使用して JBoss Web Server を起動するには:

```
# systemctl start jws5-tomcat.service
```



#### 注記

**SECURITY\_MANAGER** 変数は、アーカイブファイルのインストールに基づく JBoss Web Server 設定では非推奨になりました。次の非推奨コメントを検討してください。

```
# SECURITY_MANAGER has been deprecated. To run tomcat under the  
Java Security Manager use:  
  JAVA_OPTS="-Djava.security.manager -  
  Djava.security.policy=\\\"$CATALINA_BASE/conf/catalina.policy\\\""
```

- **systemd** を使用して JBoss Web Server を停止するには:

```
# systemctl stop jws5-tomcat.service
```

- **systemd** を使用して JBoss Web Server のステータスを確認するには:

```
# systemctl status jws5-tomcat.service
```



#### 注記

すべてのユーザーが **status** 操作を実行できます。

#### 関連情報

- RHEL 7: [System Administrator's Guide: Managing System Services](#)
- RHEL 8: [Configuring basic system settings: Managing system services with systemctl](#)
- RHEL 9: [Configuring basic system settings: Managing system services with systemctl](#)

## 2.4. コマンドラインからアーカイブのインストールを管理するための JBOSS WEB SERVER 設定

Red Hat Enterprise Linux のアーカイブファイルから JBoss Web Server をインストールする場合、コマンドラインから直接 JBoss Web Server を起動および停止できます。コマンドラインから JBoss Web Server を実行する前に、次の一連の設定タスクを実行する必要があります。

- Tomcat の **JAVA\_HOME** 環境変数を設定します。
- **tomcat** ユーザーとその親グループを作成します。
- **tomcat** ユーザーに JBoss Web Server へのアクセスを許可します。



## 注記

コマンドラインからではなく [システムデーモン](#) を使用して JBoss Web Server を管理する 場合、`.postinstall.systemd` スクリプトはこれらの設定手順を自動的に実行します。

### 2.4.1. Apache Tomcat の JAVA\_HOME 環境変数の設定

初めてコマンドラインから JBoss Web Server を実行する前に、Apache Tomcat の **JAVA\_HOME** 環境変数を設定する必要があります。

#### 前提条件

- [JBoss Web Server](#) をアーカイブファイルからインストールしました。

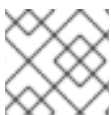
#### 手順

1. コマンドラインで、**JWS\_HOME/tomcat/bin** ディレクトリーに移動します。
2. **setenv.sh** という名前のファイルを作成します。
3. **setenv.sh** ファイルで、**JAVA\_HOME** パス定義を入力します。  
以下に例を示します。

```
export JAVA_HOME=/usr/lib/jvm/jre-1.8.0-openjdk.x86_64
```

### 2.4.2. Tomcat ユーザーとグループの作成

コマンドラインから初めて JBoss Web Server を実行する前に、**Tomcat** ユーザーアカウントとユーザーグループを作成して、シンプルかつ安全なユーザー管理を可能にする必要があります。Red Hat Enterprise Linux では、**tomcat** ユーザーのユーザー識別子 (UID) と **tomcat** グループのグループ識別子 (GID) の両方に **53** という予約値があります。



## 注記

この手順のすべてのステップを root ユーザーとして実行する必要があります。

#### 前提条件

- Tomcat の **JAVA\_HOME** 環境変数を設定しました。

#### 手順

1. コマンドラインで、**JWS\_HOME** ディレクトリーに移動します。
2. **tomcat** ユーザーグループを作成します。

```
# groupadd -g 53 -r tomcat
```

3. **tomcat** ユーザーグループに **tomcat** ユーザーを作成します。

```
# useradd -c "tomcat" -u 53 -g tomcat -s /sbin/nologin -r tomcat
```

前述のコマンドは、UID と GID の両方を **53** に設定します。後で UID と GID の値を変更する場合は、[Tomcat ユーザーとグループの UID と GID を変更する](#) を参照してください。

### 2.4.3. Tomcat ユーザーに JBoss Web Server へのアクセス権を付与する

初めてコマンドラインから JBoss Web Server を実行する前に、Tomcat ディレクトリーの所有権を **tomcat** ユーザーに割り当てて、**Tomcat** ユーザーに JBoss Web Server へのアクセスを許可する必要があります。



#### 注記

この手順のすべてのステップを root ユーザーとして実行する必要があります。

#### 前提条件

- [tomcat ユーザーとその親グループを作成しました](#)。

#### 手順

1. **JWS\_HOME** ディレクトリーに移動します。
2. Tomcat ディレクトリーの所有権を **tomcat** ユーザーに割り当てます。

```
# chown -R tomcat:tomcat tomcat/
```

3. **tomcat** ユーザーに、すべての親ディレクトリーへの実行権限があることを確認します。

```
# chmod -R u+X tomcat/
```

#### 検証

- **tomcat** ユーザーがディレクトリーの所有者であることを確認します。

```
# ls -l
```

## 2.5. アーカイブファイルからインストールした場合のコマンドラインからの JBOSS WEB SERVER の起動

Red Hat Enterprise Linux のアーカイブファイルから JBoss Web Server をインストール場合、コマンドラインから JBoss Web Server を直接起動できます。

#### 前提条件

- Tomcat の **JAVA\_HOME** 環境変数を設定しました。
- [tomcat ユーザーとその親グループを作成しました](#)。
- [tomcat ユーザーに JBoss Web Server へのアクセスを許可しました](#)。

#### 手順

- **tomcat** ユーザーとして以下のコマンドを入力します。

■

```
$ sh JWS_HOME/tomcat/bin/startup.sh
```

## 2.6. アーカイブファイルからインストールした場合にコマンドラインから JBOSS WEB SERVER を停止する

Red Hat Enterprise Linux のアーカイブファイルから JBoss Web Server をインストールする場合、コマンドラインから JBoss Web Server を直接停止できます。

### 前提条件

- コマンドラインから JBoss Web Server を起動しました。

### 手順

- **tomcat** ユーザーとして以下のコマンドを入力します。

```
$ sh JWS_HOME/tomcat/bin/shutdown.sh
```

## 2.7. JBOSS WEB SERVER の SELINUX ポリシー

Security-Enhanced Linux (SELinux) ポリシーを使用して、JBoss Web Server のアクセス制御を定義できます。これらのポリシーは、製品へのアクセス権を決定する一連のルールです。

### 2.7.1. jws5-tomcat の SELinux ポリシー情報

SELinux セキュリティモデルはカーネルにより適用され、ファイルシステムの場所やポートなどのリソースへのアプリケーションのアクセスが限定されるようにします。SELinux ポリシーは、危険にさらされているか、不適切な設定である誤ったプロセスを制限したり、実行できないようにしたりします。

JBoss Web Server インストールの **jws5-tomcat-selinux** パッケージは **jws5\_tomcat** ポリシーを提供します。次の表には、提供されている SELinux ポリシーに関する情報が含まれています。

表2.1 RPM およびデフォルトの SELinux ポリシー

名前	ポート情報	ポリシー情報
<b>jws5_tomcat</b>	<b>http_port_t</b> の 4 ポート (TCP ポート <b>8080</b> 、 <b>8005</b> 、 <b>8009</b> 、および <b>8443</b> ) の 4 つのポートにより、tomcat プロセスがそのプロセスが使用できるようになります。	<p><b>jws5_tomcat</b> ポリシーがインストールされ、Tomcat の実行時にプロセスに適した SELinux ドメインを設定します。また、適切なコンテキストを設定して、Tomcat が次のディレクトリーに書き込めるようにします。</p> <ul style="list-style-type: none"> <li>● <b>/var/opt/rh/jws5/lib/tomcat</b></li> <li>● <b>/var/opt/rh/jws5/log/tomcat</b></li> <li>● <b>/var/opt/rh/jws5/cache/tomcat</b></li> <li>● <b>/var/opt/rh/jws5/run/tomcat.pid</b></li> </ul>

## 関連情報

- Red Hat Enterprise Linux 7 での SELinux の使用に関する詳細は、[RHEL 7 SELinux User's and Administrator's Guide](#) を参照してください。
- Red Hat Enterprise Linux 8 での SELinux の使用に関する詳細は、[RHEL8 Using SELinux guide](#) を参照してください。

### 2.7.2. JBoss Web Server アーカイブインストール用の SELinux ポリシーのインストール

このリリースでは、アーカイブパッケージが SELinux ポリシーを提供します。**jws-5.7.0-application-server-`<platform>-<architecture>.zip`** アーカイブの **tomcat** フォルダーには、**.postinstall.selinux** ファイルが含まれています。必要に応じて、**.postinstall.selinux** スクリプトを実行できます。

#### 手順

1. **selinux-policy-devel** パッケージをインストールします。

```
yum install -y selinux-policy-devel
```

2. **.postinstall.selinux** スクリプトを実行します。

```
cd <JWS_home>/tomcat/  
sh .postinstall.selinux
```

3. JBoss Web Server の必要なポートにアクセスパーミッションを追加します。

```
semanage port -a -t http_port_t -p tcp <port>
```



#### 注記

JBoss Web Server は Red Hat Enterprise Linux システムのポート **8080**、**8009**、**8443**、**8005** にアクセスできます。

JBoss Web Server に追加のポートが必要な場合は、前述の **semanage** コマンドを使用して必要な権限を提供し、**<port>** を必要なポートに置き換えます。

4. Tomcat を起動します。

```
<JWS_home>/tomcat/bin/startup.sh
```

5. **jws5\_tomcat** が予想される実行中のプロセスのコンテキストを確認します。

```
ps -eo pid,user,label,args | grep jws5_tomcat | head -n1
```

6. Tomcat ディレクトリーのコンテキストを確認します。以下に例を示します。

```
ls -lZ <JWS_home>/tomcat/logs/
```



## 注記

デフォルトでは、JBoss Web Server が提供する SELinux ポリシーはアクティブではなく、Tomcat プロセスは **unconfined\_java\_t** ドメインで実行されます。このドメインは、プロセスを限定するものではありません。

提供されている SELinux ポリシーを有効にしないことを選択した場合は、次のセキュリティ対策を講じることができます。

- **tomcat** ユーザーのファイルアクセスを制限して、**Tomcat** ユーザーが JBoss Web Server ランタイムに必要なファイルとディレクトリーのみにアクセスできるようにします。
- root ユーザーで Tomcat を実行しないでください。



## 注記

JBoss Web Server がアーカイブファイルからインストールされた場合、Red Hat はネットワークファイル共有 (NFS) の使用を正式にサポートしません。JBoss Web Server インストールで NFS マウントのファイルシステムを使用する場合には、このタイプのデプロイメントに対応するために SELinux ポリシーが正しく変更されていることを確認する必要があります。

## 2.8. TOMCAT ユーザーおよびグループの UID および GID の変更

Red Hat Enterprise Linux では、**tomcat** ユーザーのユーザー識別子 (UID) と **tomcat** グループのグループ識別子 (GID) の両方に **53** という予約値があります。セットアップ要件に応じて、**tomcat** ユーザーおよびグループの UID と GID を他の値に変更できます。



### 警告

SELinux の競合を回避するには、500 未満の UID および GID 値を使用します。SELinux が **enforcing** モードに設定されている場合は、UID および GID の値が 500 を超えると、予期しない問題が発生する可能性があります。

### 前提条件

- **tomcat** ユーザーアカウントとグループが作成されました。

### 手順

1. JBoss Web Server がすでに実行されている場合は、**tomcat** ユーザーとして JBoss Web Server を停止します。詳細は、[アーカイブファイルからインストールした場合のコマンドラインからの JBoss Web Server の停止](#) を参照してください。
2. **tomcat** ユーザーとグループの現在の UID と GID を表示するには、root ユーザーとして次のコマンドを入力します。

```
id tomcat
```



前述のコマンドは、ユーザーアカウントとグループの詳細を表示します。以下に例を示します。

```
uid=53(tomcat) gid=53(tomcat) groups=53(tomcat)
```

- 新しい GID を **tomcat** グループに割り当てるには、root ユーザーとして次のコマンドを入力します。

```
groupmod -g <new_gid> tomcat
```

以下に例を示します。

```
groupmod -g 410 tomcat
```

- 新しい UID を **tomcat** ユーザーに割り当てるには、root ユーザーとして次のコマンドを入力します。

```
usermod -u <new_uid> -g <new_gid> tomcat
```

以下に例を示します。

```
usermod -u 401 -g 410 tomcat
```

- ファイルとディレクトリーのアクセス許可を新しい UID に再割り当てするには、root ユーザーとして次のコマンドを入力します。

```
# find / -not -path '/proc*' -uid <original_uid> | perl -e '$ug = @ARGV[0]; foreach $fn (<STDIN>) { chomp($fn);$m = (stat($fn))[2];chown($ug,-1,$fn);chmod($m,$fn)}' <new_uid>
```

前述のコマンドで、<original\_uid> を古い UID に置き換え、<new\_uid> を新しい UID に置き換えます。たとえば、ファイルとディレクトリーのアクセス許可を UID **53** から UID **401** に再割り当てするには、次のコマンドを入力します。

```
# find / -not -path '/proc*' -uid 53 | perl -e '$ug = @ARGV[0]; foreach $fn (<STDIN>) { chomp($fn);$m = (stat($fn))[2];chown($ug,-1,$fn);chmod($m,$fn)}' 401
```

- ファイルとディレクトリーのアクセス許可を新しい GID に再割り当てするには、root ユーザーとして次のコマンドを入力します。

```
# find / -not -path '/proc*' -gid <original_gid> | perl -e '$ug = @ARGV[0]; foreach $fn (<STDIN>) { chomp($fn);$m = (stat($fn))[2];chown(-1,$ug,$fn);chmod($m,$fn)}' <new_gid>
```

前述のコマンドで、<original\_gid> を古い GID に置き換え、<new\_gid> を新しい GID に置き換えます。たとえば、ファイルとディレクトリーのアクセス許可を GID **53** から GID **410** に再割り当てするには、次のコマンドを入力します。

```
# find / -not -path '/proc*' -gid 53 | perl -e '$ug = @ARGV[0]; foreach $fn (<STDIN>) { chomp($fn);$m = (stat($fn))[2];chown(-1,$ug,$fn);chmod($m,$fn)}' 410
```

- JBoss Web Server を **tomcat** ユーザーとして再起動するには、[アーカイブファイルからインストールされた場合のコマンドラインからの JBoss Web Server の起動](#) を参照してください。

## 関連情報

- [Red Hat Enterprise Linux で予約されている UID/GID とは何ですか？](#)

## 第3章 RPM パッケージからの RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER のインストール

アーカイブファイルまたは RPM パッケージから Red Hat Enterprise Linux に JBoss Web Server をインストールできます。RPM パッケージから JBoss Web Server をインストールする場合、インストールパッケージは Red Hat Subscription Management から入手できます。RPM インストールオプションは、Red Hat Enterprise Linux バージョン 7、8、および 9 で使用できます。

RPM パッケージから JBoss Web Server をインストールすると、Tomcat がサービスとしてデプロイされ、Tomcat リソースが絶対パスにインストールされます。

### 3.1. 前提条件

- YUM パッケージを使用するか、圧縮されたアーカイブから、サポートされている Java Development Kit (JDK) をインストールしている。
- 使用中のシステムが Red Hat Enterprise Linux パッケージ要件に準拠している。

#### 3.1.1. YUM パッケージマネージャーを使用した JDK のインストール

YUM パッケージマネージャーを使用して、Java Development Kit (JDK) をインストールできます。サポートされている JDK の完全なリストについては、[JBoss Web Server operating systems and configurations](#) を参照してください。

#### 手順

1. Red Hat Enterprise Linux システムを適切なチャンネルにサブスクライブします。

- **OpenJDK:**
  - rhel-7-server-rpms
  - rhel-8-server-rpms
  - rhel-9-server-rpms
- **IBM:**
  - rhel-7-server-supplementary-rpms
  - rhel-8-server-supplementary-rpms
  - rhel-9-server-supplementary-rpms



#### 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

2. root ユーザーとしてコマンドを実行し、1.8 JDK をインストールします。

```
# yum install java-1.8.0-<VENDOR>-devel
```

<VENDOR> を **ibm** または **openjdk** に置き換えます。

3. root ユーザーとして以下のコマンドを実行し、正しい JDK が使用されていることを確認します。

```
# alternatives --config java
```

```
# alternatives --config javac
```

これらのコマンドは、利用可能な JDK バージョンと、プラス (+) 記号の付いた選択したバージョンのリストを返します。選択した JDK が必要な JDK ではない場合は、シェルプロンプトで指示されたように、必要な JDK に変更します。



### 重要

**java** コマンドおよび **javac** コマンドを使用するすべてのソフトウェアは、**alternatives** によって設定された JDK を使用します。Java の代替の変更は、他のソフトウェアの実行に影響を及ぼす可能性があります。

## 3.1.2. 圧縮アーカイブからの JDK のインストール

**.zip** または **.tar** ファイルなどの圧縮アーカイブから Java Development Kit (JDK) をインストールできます。サポートされている JDK の完全なリストについては、[JBoss Web Server operating systems and configurations](#) を参照してください。

### 手順

1. JDK がベンダーの Web サイト (Oracle または OpenJDK) からダウンロードされている場合は、ベンダーが提供するインストール手順を使用して、**JAVA\_HOME** 環境変数を設定します。
2. JDK が圧縮されたアーカイブからインストールされた場合は、Tomcat の **JAVA\_HOME** 環境変数を設定します。
  - a. Tomcat の **bin** ディレクトリー (**JWS\_HOME/tomcat/bin**) で、**setenv.sh** という名前のファイルを作成します。
  - b. **setenv.sh** ファイルで、**JAVA\_HOME** パス定義を入力します。以下に例を示します。

```
$ cat JWS_HOME/tomcat/bin/setenv.sh
```

```
export JAVA_HOME=/usr/lib/jvm/jre-1.8.0-openjdk.x86_64
```

## 3.1.3. Red Hat Enterprise Linux パッケージの要件

Red Hat Enterprise Linux に JBoss Web Server をインストールする前に、システムが以下のパッケージ要件に準拠していることを確認する必要があります。



### 注記

パッケージの要件は、使用している Red Hat Enterprise Linux のバージョンによって異なります。

- Red Hat Enterprise Linux バージョン 8 または 9 で、OpenSSL または Apache Portable Runtime (APR) を使用する場合は、Red Hat Enterprise Linux が提供する **openssl** および **apr** パッケージをインストールする必要があります。

- **openssl** パッケージをインストールするには、root ユーザーとして次のコマンドを入力します。

```
# yum install openssl
```

- **apr** パッケージをインストールするには、root ユーザーとして次のコマンドを入力します。

```
# yum install apr
```

- **tomcat-native** パッケージをインストールする前に、**tomcatjss** パッケージを削除する必要があります。**tomcatjss** パッケージは、OpenSSL セキュリティーモデルではなく、基盤となる Network Security Services (NSS) セキュリティーモデルを使用します。**tomcatjss** パッケージを削除するには、root ユーザーとして次のコマンドを入力します。

```
# yum remove tomcatjss
```

### 注記

- Red Hat Enterprise Linux 7 では、JBoss Web Server は、Red Hat JBoss Core Services で提供される **openssl** および **apr** パッケージを使用します。
- Red Hat Enterprise Linux バージョン 8 および 9 では、JBoss Web Server は **openssl** および **apr** パッケージを提供しません。JBoss Web Server は、Red Hat Enterprise Linux が提供する openssl および apr パッケージを使用します。Red Hat Enterprise Linux バージョン 8 または 9 で OpenSSL または APR を使用する場合は、このセクションで前述したように、オペレーティングシステムから openssl および apr パッケージをインストールする必要があります。

## 3.2. RED HAT ENTERPRISE LINUX へのサブスクリプションの割り当て

JBoss Web Server の RPM パッケージをダウンロードしてインストールする前に、Red Hat Subscription Management にシステムを登録し、該当するコンテンツ配信ネットワーク (CDN) リポジトリをサブスクライブする必要があります。その後、いくつかの検証手順を実行して、サブスクリプションが必要な CDN リポジトリを提供していることを確認することができます。

### 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

### 手順

1. Red Hat [Subscription Management](#) Web ページにログインします。
2. **System** タブをクリックします。
3. サブスクリプションを追加するシステムの **Name** をクリックします。
4. **Details** タブから **Subscriptions** タブに移動し、**Attach Subscriptions** をクリックします。
5. アタッチするサブスクリプションの横にあるチェックボックスをオンにしてから、**Attach Subscriptions** をクリックします。

## 検証

1. Red Hat [サブスクリプション](#) Web ページにログインします。
2. **Subscription Name** 列で、選択するサブスクリプションをクリックします。
3. **Products Provided** では、次の両方が必要です。
  - **JBoss Enterprise Web Server**
  - **Red Hat JBoss Core Services**

## 関連情報

- [RHEL 7: Installation Guide: Subscription Manager](#)
- [RHEL 8: Performing a Standard RHEL 8 Installation: Registering your system using the Subscription Manager User Interface](#)
- [RHEL 9: Performing a Standard RHEL 9 Installation: Registering your system using the Subscription Manager User Interface](#)

## 3.3. YUMを使用した RPM パッケージからの JBOSS WEB SERVER のインストール

YUM パッケージマネージャーを使用して、JBoss Web Server を RPM パッケージから Red Hat Enterprise Linux にインストールできます。

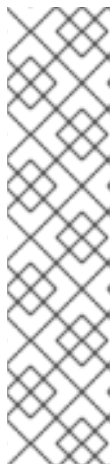
### 前提条件

- [サポート対象の Java Development Kit \(JDK\)](#) がインストールされている。
- 使用中のシステムが [Red Hat Enterprise Linux パッケージ要件](#) に準拠している。
- [Red Hat Enterprise Linux](#) にサブスクリプションをアタッチしている。

### 手順

1. ご使用のオペレーティングシステムバージョンの JBoss Web Server CDN リポジトリにサブスクライブするには、次のコマンドを入力します。

```
# subscription-manager repos --enable <repository>
```



## 注記

上記のコマンドで、**<repository>** を次の値に置き換えます。

- Red Hat Enterprise Linux 7 で、**<repository>** を **jws-5-for-rhel-7-server-rpms** および **jb-coreservices-1-for-rhel-7-server-rpms** の両方に置き換えます。
- Red Hat Enterprise Linux 9 では、**<repository>** を **jws-6-for-rhel-9-x86\_64-rpms** に置き換えます。
- Red Hat Enterprise Linux 9 では、**<repository>** を **jws-6-for-rhel-9-x86\_64-rpms** に置き換えます。

2. JBoss Web Server をインストールするには、root ユーザーとして次のコマンドを入力します。

```
# yum groupinstall jws5
```



## 重要

RPM パッケージから JBoss Web Server をインストールする場合、**JWS\_HOME** フォルダは **/opt/rh/jws5/root/usr/share** です。



## 注記

- **groupinstall** コマンドを使用するのではなく、各パッケージとその依存関係を個別にインストールできます。推奨される方法は、**groupinstall** を使用することです。
- Red Hat JBoss Core Services リポジトリは、JBoss Web Server を Red Hat Enterprise Linux 7 にインストールする場合にのみ必要です。Red Hat Enterprise Linux バージョン 8 または 9 に JBoss Web Server をインストールする場合、Red Hat JBoss Core Services リポジトリは **必要ありません**。
- Software Collection を使用して NFS の使用を有効にする機能が有効になっています。この機能の詳細については、[Packaging Guide: Using Software Collections over NFS](#) を参照してください。

## 3.4. RPM からインストールした場合の JBOSS WEB SERVER の起動

RPM パッケージから JBoss Web Server をインストールすると、コマンドラインを使用して JBoss Web Server を起動できます。その後、service **status** コマンドの出力を表示して、Tomcat が正常に実行されていることを確認できます。



## 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

## 手順

- root ユーザーとして以下のコマンドを実行します。

```
# systemctl start jws5-tomcat.service
```



### 注記

これは、RPM インストールに JBoss Web Server を起動するのにサポートされる唯一の方法です。

### 検証

- Tomcat が実行されていることを確認するには、任意のユーザーとして次のコマンドを入力します。

```
# systemctl status jws5-tomcat.service
```

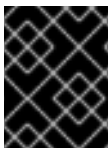


### 注記

RHEL 8 での HTTPD のインストールと設定の詳細については、[Deploying Different Types of Server: Setting up the Apache HTTP web server](#) を参照してください。

## 3.5. RPM からインストールした場合の JBOSS WEB SERVER の停止

RPM パッケージから JBoss Web Server をインストールすると、コマンドラインを使用して JBoss Web Server を停止できます。その後、service **status** コマンドの出力を表示して、Tomcat が正常に実行されていることを確認できます。



### 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

### 手順

- root ユーザーとして次のコマンドを入力します。

```
# systemctl stop jws5-tomcat.service
```

### 検証

- Tomcat が実行されていないことを確認するには、任意のユーザーとして次のコマンドを入力します。

```
# systemctl status jws5-tomcat.service
```



### 注記

RHEL 8 での HTTPD のインストールと設定の詳細については、[Deploying Different Types of Server: Setting up the Apache HTTP web server](#) を参照してください。

## 3.6. システムの起動時に JBOSS WEB SERVER サービスを開始するように設定する



RPM パッケージから JBoss Web Server をインストールすると、JBoss Web Server サービスがシステム起動時に開始するように設定できます。

## 手順

- 以下のコマンドを入力します。

```
# systemctl enable jws5-tomcat.service
```



### 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

## 3.7. JBOSS WEB SERVER の SELINUX ポリシー

Security-Enhanced Linux (SELinux) ポリシーを使用して、JBoss Web Server のアクセス制御を定義できます。これらのポリシーは、製品へのアクセス権を決定する一連のルールです。

### 3.7.1. jws5-tomcat の SELinux ポリシー情報

SELinux セキュリティーモデルはカーネルにより適用され、ファイルシステムの場所やポートなどのリソースへのアプリケーションのアクセスが限定されるようになります。SELinux ポリシーは、危険にさらされているか、不適切な設定である誤ったプロセスを制限したり、実行できないようにしたりします。

JBoss Web Server インストールの **jws5-tomcat-selinux** パッケージは **jws5\_tomcat** ポリシーを提供します。次の表には、提供されている SELinux ポリシーに関する情報が含まれています。

表3.1 RPM およびデフォルトの SELinux ポリシー

名前	ポート情報	ポリシー情報
<b>jws5_tomcat</b>	<b>http_port_t</b> の 4 ポート (TCP ポート <b>8080</b> 、 <b>8005</b> 、 <b>8009</b> 、および <b>8443</b> ) の 4 つのポートにより、tomcat プロセスがそのプロセスが使用できるようになります。	<p><b>jws5_tomcat</b> ポリシーがインストールされ、Tomcat の実行時にプロセスに適した SELinux ドメインを設定します。また、適切なコンテキストを設定して、Tomcat が次のディレクトリーに書き込めるようになります。</p> <ul style="list-style-type: none"> <li>• <b>/var/opt/rh/jws5/lib/tomcat</b></li> <li>• <b>/var/opt/rh/jws5/log/tomcat</b></li> <li>• <b>/var/opt/rh/jws5/cache/tomcat</b></li> <li>• <b>/var/opt/rh/jws5/run/tomcat.pid</b></li> </ul>

## 関連情報

- Red Hat Enterprise Linux 7 での SELinux の使用に関する詳細は、[RHEL 7 SELinux User's and Administrator's Guide](#) を参照してください。

- Red Hat Enterprise Linux 8 での SELinux の使用に関する詳細は、[RHEL8 Using SELinux guide](#) を参照してください。

### 3.7.2. JBoss Web Server RPM インストールの SELinux ポリシーの有効化

RPM パッケージから JBoss Web Server をインストールすると、**jws5-tomcat-selinux** パッケージが JBoss Web Server の SELinux ポリシーを提供します。これらのパッケージは、JBoss Web Server チャネルで入手できます。

#### 手順

1. **jws5-tomcat-selinux** パッケージをインストールします。

```
yum install -y jws5-tomcat-selinux
```

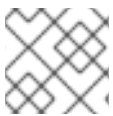
## 第4章 MICROSOFT WINDOWS への JBOSS WEB SERVER のインストール

[Red Hat カスタマーポータル](#) からダウンロードできる一連のアーカイブファイルから Microsoft Windows に JBoss Web Server をインストールできます。

### 4.1. MICROSOFT WINDOWS への JDK のインストール

JBoss Web Server を Microsoft Windows にインストールする前に、まず Java Development Kit (JDK) をインストールする必要があります。

Oracle や IBM など、サポートされているベンダーの Web サイトから JDK をダウンロードしてインストールできます。サポートされている JDK のリストについては、[サポートされているオペレーティングシステムと設定](#) を参照してください。



#### 注記

この手順では、Oracle JDK をインストールする方法について説明します。

#### 手順

1. Oracle Web サイトにアクセスするには、ブラウザーウィンドウを開き、次の URL を入力します。  
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>
2. オペレーティングシステムおよびアーキテクチャー用の Oracle JDK をダウンロードします。
3. ダウンロードしたファイルをダブルクリックしてインストールを開始します。
4. インストールウィンドウで指示どおりに続行します。

#### 関連情報

- IBM JDK のインストールについては、[IBM 開発者キット](#) の Web ページを参照してください。

### 4.2. MICROSOFT WINDOWS での JBOSS WEB SERVER のダウンロードと抽出

JBoss Web Server のアーカイブファイルは、Red Hat Customer Portal の [製品ダウンロード](#) ページからダウンロードできます。

#### 前提条件

- [サポートされているベンダーの Web サイト](#) から Java 開発キット (JDK) をダウンロードしてインストールしました。

#### 手順

1. ブラウザーを開き、Red Hat [製品ダウンロード](#) ページにログインします。
2. **Product Downloads** リストの **Red Hat JBoss Web Server** をクリックします。

3. Software Downloads ページで、**Version** ドロップダウンメニューから正しい JBoss Web Server バージョンを選択します。
4. ファイルのダウンロードテーブルで、次の各ファイルの横にある **Download** をクリックします。
  - The Red Hat JBoss Web Server 5.5 Application Server (**jws-5.5.0-application-server.zip**).
  - Red Hat JBoss Web Server 5.5 Native Components for Windows Server (**jws-5.5.0-application-server-<platform>-<architecture>.zip**)



#### 注記

システムのプラットフォームとアーキテクチャーに一致する正しいファイルを選択してください。

5. ダウンロードしたアーカイブファイルをインストールディレクトリーにデプロイメントします。

アーカイブを解凍すると、JBoss Web Server の最上位ディレクトリーが作成されます。このドキュメントでは、JBoss Web Server の最上位ディレクトリーを **JWS\_HOME** と呼んでいます。

## 4.3. MICROSOFT WINDOWS での JBOSS WEB SERVER の設定

JBoss Web Server を Microsoft Windows にインストールすると、コマンドプロンプトから、またはコンピューター管理ツールを使用して JBoss Web Server を管理できます。

Microsoft Windows で JBoss Web Server を実行する前に、次の一連の設定タスクを実行する必要があります。

- [環境変数の設定](#)
- [Tomcat サービスをインストールします。](#)
- [フォルダーのアクセス許可を設定します。](#)

### 4.3.1. Microsoft Windows での JBoss Web Server の環境変数の設定

Microsoft Windows で初めて JBoss Web Server を実行する前に、**JAVA\_HOME**、**TMP**、および **TEMP** 環境変数を設定する必要があります。**PATH** 環境変数も更新する必要があります。

#### 前提条件

- [JBoss Web Server がインストール](#) されました。

#### 手順

1. ローカル管理者権限を持つアカウントにログインします。
2. **Control Panel > System** をクリックします。
3. **Advanced** タブをクリックします。
4. **環境変数** ボタンをクリックします。

5. システム変数の **新規** ボタンをクリックします。
6. **JAVA\_HOME**、**TMP**、**TEMP** に、システムに適切な名前と値のペアを入力します。
7. SSL コネクターが正常に機能するようにするには、サービスを実行するユーザーの **PATH** 環境変数に **JWS\_HOME \bin** を追加します。



#### 注記

デフォルトでは、サービスは **SYSTEM** ユーザーの下で実行されます。

### 4.3.2. Microsoft Windows への Tomcat サービスのインストール

Microsoft Windows で初めて JBoss Web Server を実行する前に、Tomcat サービスをインストールする必要があります。

#### 前提条件

- [JBoss Web Server の環境変数を設定](#) しました。

#### 手順

1. 管理者権限でコマンドプロンプトを開き、Tomcat バージョンの **bin** フォルダーに移動します。

```
cd /D "JWS_HOME\tomcat\bin"
```

2. Tomcat サービスをインストールします。

```
call service.bat install
```

### 4.3.3. Microsoft Windows での JBoss Web Server サービスのフォルダー権限の設定

Microsoft Windows で初めて JBoss Web Server を実行する前に、JBoss Web Server サービスのフォルダー権限を設定する必要があります。フォルダーのパーミッションを設定すると、JBoss Web Server サービスの実行に使用されるアカウントが、**JWS\_HOME** フォルダーとそのすべてのサブフォルダーを完全に制御できるようになります。

#### 前提条件

- [Microsoft Windows に Tomcat サービスをインストール](#) しました。

#### 手順

1. **JWS\_HOME** ディレクトリーを右クリックし、**Properties** をクリックします。
2. **Security** タブを選択します。
3. **Edit** ボタンをクリックします。
4. **Add** ボタンをクリックします。
5. テキストボックスに **LOCAL SERVICE** を入力します。

6. **LOCAL SERVICE** アカウントの **Full Control** チェックボックスを選択します。
7. **OK** をクリックします。
8. **Advanced** ボタンをクリックします。
9. **Advanced Security Settings** ダイアログ内で **LOCAL SERVICE** を選択し、**Edit** をクリックします。
10. **Replace all existing inheritable permissions on all descendants with inheritable permissions from this object** オプションの横にあるチェックボックスを選択します。
11. 開いているすべてのフォルダープロパティウィンドウで **OK** をクリックして設定を適用します。

## 4.4. MICROSOFT WINDOWS での JBOSS WEB SERVER の起動

JBoss Web Server を Microsoft Windows にインストールすると、コマンドプロンプトまたはコンピューター管理ツールを使用して Tomcat サービスを開始できます。

### 前提条件

- [JBoss Web Server を設定](#) しました。

### 手順

- 以下のいずれかの手順を実行します。
  - 管理者としてコマンドプロンプトを開き、以下のコマンドを入力します。
 

```
net start tomcat9
```
  - **Start > Administrative Tools > Services** をクリックし、**Tomcat9** サービスを右クリックしてから、**Start** をクリックします。

### 注記

一部のサードパーティーアプリケーションは、Windows のシステムディレクトリーにライブラリーを追加します。これらのサードパーティーライブラリーは、検索時に Tomcat ライブラリーよりも優先されます。サードパーティーライブラリーが Tomcat ネイティブライブラリーと同じ名前の場合、システムは JBoss Web Server で配布されるライブラリーではなく、サードパーティーライブラリーをロードします。この状況では、Tomcat が正常に起動しない可能性があり、Tomcat は Windows イベントログまたは Tomcat ログファイルにエラーメッセージを記録しません。

この現象が発生した場合は、次の手順を実行できます。

- エラーを表示するには、**catalina.bat run** コマンドを実行します。
- **C:\windows\System32\**ディレクトリーとその他の **PATH** ディレクトリーの内容を調べます。
- ダイナミックリンクライブラリー (DLL) が JBoss Web Server ライブラリーと競合しないようにしてください。特に、**libeay32.dll**、**ssleay32.dll**、および **libssl32.dll** ライブラリーを探します。

## 4.5. MICROSOFT WINDOWS での JBOSS WEB SERVER の停止

JBoss Web Server を Microsoft Windows にインストールすると、コマンドプロンプトまたはコンピューター管理ツールを使用して Tomcat サービスを停止できます。

### 前提条件

- [Microsoft Windows で JBoss Web Server を起動](#) しました。

### 手順

- 以下のいずれかの手順を実行します。
  - 管理者としてコマンドプロンプトを開き、以下のコマンドを入力します。

```
net stop tomcat9
```
  - **Start > Administrative Tools > Services**に移動し、**Tomcat9** サービスを右クリックして、**Stop** をクリックします。

## 第5章 JBOSS WEB SERVER 用の HIBERNATE の設定

Hibernate Object/Relational Mapping (ORM) は、JBoss Web Server を Java データベース接続 (JDBC) データソースに接続するために使用できるオブジェクトリレーショナルマッピングフレームワークです。アプリケーションデータをリレーショナルデータベースに格納する場合は、JBoss Web Server で Hibernate ORM を使用できます。

### 5.1. HIBERNATE ORM のインストール

JBoss Web Server がサポートするすべてのプラットフォームに Hibernate ORM をインストールできます。Hibernate JAR ファイルは、次の 2 つの方法のいずれかで取得できます。

- [Red Hat カスタマーポータル](#) から **jboss-web-server-5.7.0-maven-repository.zip** としてダウンロードできる JBoss Web Server Maven リポジトリを使用します。
- <https://maven.repository.redhat.com/ga/> でアクセスできる、Red Hat がホストする Maven リポジトリを使用します。



#### 注記

Red Hat は、今後のリリースで Hibernate JAR ファイルを **jboss-web-server-X.X.x-maven-repository.zip** から削除する予定です。JBoss Web Server 5.7 リリースでは、Red Hat は引き続き **jboss-web-server-5.7.0-maven-repository.zip** で Hibernate JAR ファイルを提供しますが、Red Hat はこれらの JAR ファイルに対して、セキュリティ修正の可能性以外の更新を提供することはありません。

JBoss Web Server 5.7 および今後のバージョンでは、Red Hat は <https://maven.repository.redhat.com/ga/> でアクセスできる Red Hat がホストする Maven リポジトリで Hibernate JAR ファイルを引き続き提供および維持します。

#### 前提条件

- 次の Maven リポジトリのいずれかを使用するようにプロジェクトを設定している。
  - JBoss Web Server Maven リポジトリ (**jboss-web-server-5.7.0-maven-repository.zip**)
  - Red Hat がホストする Maven リポジトリ (<https://maven.repository.redhat.com/ga/>)

#### 手順

1. 使用する Maven リポジトリに応じて、**jboss-web-server-X.X.x-maven-repository.zip** または <https://maven.repository.redhat.com/ga/> から Hibernate JAR ファイルを取得します。
2. Hibernate JAR ファイルをデプロイメント WAR ファイルに追加します。

#### 関連情報

- [JBoss Web Server ドキュメントの Hibernate](#)

### 5.2. JDBC 接続プールの設定

Apache Tomcat は JDBC データソースのデフォルトの接続プールメカニズムを提供します。JBoss Web Server デプロイメントの **/META-INF/context.xml** ファイルの設定を更新することで、JDBC 接続プールを設定できます。



## 手順

1. **/META-INF/context.xml** ファイルを開きます。
2. アプリケーションで使用できる JDBC 接続プールを変更します。  
以下に例を示します。

```
<Context>
  <Resource
    name="jdbc/DsWebAppDB"
    auth="Container"
    type="javax.sql.DataSource"
    username="sa"
    password=""
    driverClassName="org.h2.Driver"
    url="jdbc:h2:mem:target/test/db/h2/hibernate"
    maxActive="8"
    maxIdle="4"/>
</Context>
```

## 5.3. HIBERNATE 接続プロパティの設定

JBoss Web Server デプロイメントの **/WEB-INF/classes/META-INF/persistence.xml** ファイルの設定を更新することで、Tomcat プールからの JDBC 接続を使用するように Hibernate を設定できます。



## 注記

Hibernate API を直接使用する場合は、**hibernate.cfg.xml** ファイルと同様の設定を使用します。

## 手順

1. **/WEB-INF/classes/META-INF/persistence.xml** ファイルを開きます。
2. Tomcat からの接続を使用するように Hibernate を設定します。  
以下に例を示します。

```
<persistence version="1.0"
  xmlns="http://java.sun.com/xml/ns/persistence"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
  http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd">

  <persistence-unit name="dswebapp">
    <provider>org.hibernate.ejb.HibernatePersistence</provider>
    <properties>
      <property name="hibernate.dialect" value="org.hibernate.dialect.H2Dialect" />
      <property name="hibernate.connection.datasource"
value="java:comp/env/jdbc/DsWebAppDB"/>
    </properties>
  </persistence-unit>
</persistence>
```

## 5.4. JDBC データソースの追加

JBoss Web Server デプロイメントの **/WEB-INF/web.xml** ファイルの設定を更新することで、JDBC データソースを消費するように Tomcat を設定できます。

### 手順

1. **/WEB-INF/web.xml** ファイルを開きます。
2. **resource-env-ref** 要素を使用して JDBC データソースを設定します。  
以下に例を示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">

  <resource-env-ref>
    <resource-env-ref-name>jdbc/DsWebAppDB</resource-env-ref-name>
    <resource-env-ref-type>javax.sql.DataSource</resource-env-ref-type>
  </resource-env-ref>
</web-app>
```



### 注記

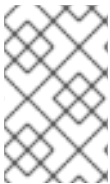
前の例では、**jdbc/DsWebAppDB** データソースを使用しています。

## 第6章 RED HAT JBOSS WEB SERVER の HTTP/2 の有効化

Hypertext Transfer Protocols (HTTP) は、インターネットを介して (サーバーやブラウザーなどの) アプリケーション間でデータを送信する標準的な方法です。JBoss Web Server は、Transport Layer Security (TLS) を使用している暗号化接続のための HTTP/2 の使用をサポートしており、これは有効になっている場合に **h2** キーワードで示されます。

HTTP/2 は、以下のような機能強化を提供し、HTTP/1.1 よりも改良されています。

- ヘッダー圧縮は、送信されるヘッダーのサイズを小さくするために、暗黙的な情報を省略します。
- 1つの接続で複数の要求および応答がある場合、テキストフレームではなくバイナリーフレームを使用して応答メッセージを分割します。



### 注記

JBoss Web Server は、Transmission Control Protocol (TCP) を使用する暗号化されていない接続に対する HTTP/2 の使用をサポートしません。これは、有効になっている場合、**h2c** キーワードで示されます。

### 6.1. 前提条件

- Red Hat Enterprise Linux の root ユーザーアクセス権を持っている。
- Red Hat JBoss Web Server 5.0 以降がインストールされている。
- Red Hat Enterprise Linux で提供される **openssl** および **apr** パッケージがインストールされている。**openssl** および **apr** パッケージのインストールに関する詳細は、[Red Hat Enterprise Linux パッケージ要件](#) を参照してください。



### 注記

これらのオペレーティングシステムネイティブライブラリーは、利用可能な場合は **jws-5.7.0-application-server-<platform>-<architecture>.zip** から提供されます。

Red Hat Enterprise Linux バージョン 8 または 9 で JSSE+OpenSSL または APR を動作させる場合は、正常に動作するように Tomcat-Native を使用する必要があります。Tomcat-Native は、ネイティブアーカイブディレクトリーにあります。

- SSL が有効な HTTP/2 プロトコルをサポートするコネクタを設定しました。JBoss Web Server 5.7 では、次のコネクタが HTTP/2 プロトコルをサポートします。
  - APR ネイティブコネクタ (APR)
  - JSSE + OpenSSL (JSSE) のある NIO コネクタ
  - JSSE + OpenSSL (JSSE) のある NIO2 コネクタ

### 6.2. コネクタの HTTP/2 を有効にする

**server.xml** 設定ファイルの設定を更新することで、コネクターの HTTP/2 を有効にすることができます。

## 手順

1. **JWS\_HOME/tomcat/conf/server.xml** 設定ファイルを開きます。
2. **server.xml** ファイルで、HTTP/2 アップグレードプロトコルをコネクターに追加します。以下に例を示します。

```
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
  maxThreads="150" SSLEnabled="true">
  <UpgradeProtocol className="org.apache.coyote.http2.Http2Protocol" />
  <SSLHostConfig>
    <Certificate certificateKeystoreFile="/KeyStore.jks"
      certificateKeystorePassword="changeit"
      type="RSA" />
  </SSLHostConfig>
</Connector>
```

**server.xml** ファイルには、アップグレードプロトコルが HTTP/2 に設定された APR プロトコルのコネクター定義の例が含まれています。

以下に例を示します。

```
<Connector port="8443"
  protocol="org.apache.coyote.http11.Http11AprProtocol"
  maxThreads="150" SSLEnabled="true" >
  <UpgradeProtocol className="org.apache.coyote.http2.Http2Protocol" />
  <SSLHostConfig>
    <Certificate certificateKeyFile="conf/localhost-rsa-key.pem"
      certificateFile="conf/localhost-rsa-cert.pem"
      certificateChainFile="conf/localhost-rsa-chain.pem"
      type="RSA" />
  </SSLHostConfig>
</Connector>
```

3. 設定の更新を適用するには、root ユーザーとして Red Hat JBoss Web Server を再起動します。
  - **systemd** を使用して Red Hat Enterprise Linux で JBoss Web Server を再起動するには、次のコマンドを入力します。

```
# systemctl restart jws5-tomcat.service
```

- **startup.sh** を使用して Red Hat Enterprise Linux で JBoss Web Server を再起動するには、次のコマンドを入力します。

```
# JWS_HOME/sbin/shutdown.sh
# JWS_HOME/sbin/startup.sh
```

- Microsoft Windows で JBoss Web Server を再起動するには、次のコマンドを入力します。

```
# net restart tomcat9
```



## 重要

Red Hat Enterprise Linux 6 はサポートされなくなり、その後ドキュメントから削除されました。

## 6.3. JBOSS WEB SERVER ログを表示して、HTTP/2 が有効になっていることを確認する

JBoss Web Server コンソールの出力ログを表示して、HTTP/2 が有効になっていることを確認できます。

### 前提条件

- [コネクターに対して HTTP/2 を有効にしました。](#)

### 手順

- コンソール出力ログを表示するには、次のコマンドを入力します。

```
$ cat JWS_HOME/tomcat/logs/catalina.out | grep 'h2'
```



## 注記

上記のコマンドで、**JWS\_HOME** を JBoss Web Server インストールの最上位ディレクトリーに置き換えます。

### 検証

- HTTP/2 が有効になっている場合、コマンドは次のタイプの出力を生成します。これは、コネクターが **h2** へのネゴシエーションをサポートするように設定されていることを示します。

```
06-Apr-2018 04:49:26.201 INFO [main]
org.apache.coyote.http11.AbstractHttp11Protocol.configureUpgradeProtocol The ["https-
openssl-apr-8443"] connector has been configured to support negotiation to [h2] via ALPN
```

## 6.4. CURL コマンドを使用して HTTP/2 が有効になっていることを確認する

**curl** コマンドラインツールを使用して、HTTP/2 が有効化されていることを確認できます。

### 前提条件

- [コネクターに対して HTTP/2 を有効にしました。](#)
- HTTP2 をサポートするバージョンの **curl** を使用している。  
HTTP/2 をサポートするバージョンの **curl** を使用していることを確認するには、次のコマンドを入力します。

```
$ curl -V
```

このコマンドは、次のタイプの出力を生成します。

■

```
curl 7.55.1 (x86_64-redhat-linux-gnu) ...  
Release-Date: 2017-08-14  
Protocols: dict file ftp ftps gopher http https ...  
Features: AsynchDNS IDN IPv6 Largefile GSS-API Kerberos SPNEGO NTLM NTLM_WB  
SSL libz TLS-SRP HTTP2 UnixSockets HTTPS-proxy Metalink PSL
```

## 手順

1. HTTP/2 プロトコルが有効であることを確認するには、以下のコマンドを入力します。

```
$ curl -I http://<JBoss_Web_Server>:8080/
```



### 注記

前の例で、<JBoss\_Web\_Server> を **example.com** などの変更されたコネクターの URI に置き換えます。ポート番号は設定によって異なります。

## 検証

- HTTP/2 プロトコルが有効な場合、**curl** コマンドは次のような出力をします。

```
HTTP/2 200
```

そうでなければ、HTTP/2 プロトコルが無効の場合、**curl** コマンドは次のような出力をします。

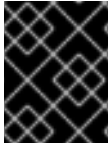
```
HTTP/1.1 200
```

## 6.5. 関連情報 (または次の手順)

- HTTP/2 の使用に関する詳細は、[Apache Tomcat 9 Configuration Reference: The HTTP Connector - HTTP/2 Support](#) を参照してください。
- HTTP/2 アップグレードプロトコルおよびサポートされる属性の詳細は、[Apache Tomcat 9 Configuration Reference: The HTTP2 Upgrade Protocol](#) を参照してください。
- HTTP/2 向けに提案されるインターネット標準の詳細については、[IETF: RFC 7540 - Hypertext Transfer Protocol Version 2 \(HTTP/2\)](#) を参照してください。

## 第7章 RED HAT JBOSS WEB SERVER でのパスワード VAULT の使用

**tomcat-vault** という名前の JBoss Web Server パスワード Vault は、Apache Tomcat の PicketLink Vault 拡張機能です。パスワード vault を使用して、パスワードやその他の機密性の高い文字列をマスクし、暗号化された Java キーストアに機密情報を格納できます。パスワード vault を使用すると、Tomcat 設定ファイルにクリアテキストのパスワードを保存するのを停止できます。Tomcat は、パスワード vault を使用して、パスワードやその他の機密性の高い文字列をキーストアから検索できます。



### 重要

パスワード vault で **CRYPT** 機能を使用する方法の詳細については、[CRYPT の使用](#) を参照してください。



### 注記

FIPS (Federal Information Processing Standard) 140-2 は、**tomcat-vault** が提供するパスワードベースの暗号化をサポートしません。JBoss Web Server ホストでパスワードベースの暗号化を使用する場合は、FIPS が無効になっていることを確認する必要があります。FIPS モードが有効な場合に **tomcat-vault** を使用しようとする、次のエラーメッセージが表示されます: **Security Vault can't be used in FIPS mode**

### 7.1. アーカイブファイルからのパスワード VAULT のインストール

JBoss Web Server をアーカイブファイルからインストールする場合、**jws-5.7.0-application-server.zip** ファイルをインストールすると、パスワード vault が自動的にインストールされます。password vault は **JWS\_HOME/tomcat/lib/tomcat-vault.jar** ファイルにあります。

### 7.2. YUM パッケージマネージャーを使用して RHEL にパスワード VAULT をインストールする

RPM パッケージから Red Hat Enterprise Linux に JBoss Web Server をインストールする場合、YUM パッケージマネージャーを使用してパスワード vault をインストールできます。

#### 手順

- root ユーザーとして以下のコマンドを実行します。

```
yum install jws5-tomcat-vault
```

### 7.3. JBOSS WEB SERVER でパスワード VAULT を有効にする

**catalina.properties** ファイルに設定プロパティを追加することで、パスワード vault を有効化できます。

#### 前提条件

- [アーカイブファイルから](#)、または [YUM パッケージマネージャーを使用](#) してパスワード vault をインストールしました。

#### 手順

1. Tomcat がすでに実行されている場合は停止します。
2. **JWS\_HOME/tomcat/conf/catalina.properties** ファイルを開きます。
3. **catalina.properties** ファイルに、次の行を入力します。

```
org.apache.tomcat.util.digester.PROPERTY_SOURCE=org.apache.tomcat.vault.util.PropertySourceVault
```



#### 注記

上記の例では、**JWS\_HOME** を JBoss Web Server インストールへのパスに置き換えます。この例に示されているパスでは、ディレクトリー区切り文字にスラッシュ (/) を使用しています。

## 7.4. JBOSS WEB SERVER での JAVA キーストアの作成

パスワード vault を使用する前に、まず **keytool -genseckey** コマンドを使用して Java キーストアを作成する必要があります。

#### 手順

- 以下のコマンドを入力します。

```
$ keytool -genseckey \
-keystore JWS_HOME/tomcat/vault.keystore \
-alias my_vault \
-storetype jceks \
-keyalg AES \
-keysize 128 \
-storepass <vault_password> \
-keypass <vault_password> \
-validity 730
```



#### 注記

前の例では、パラメーター設定を環境に適した値に置き換えます。

各パラメーターの詳細については、**keytool -genseckey -help** コマンドを使用してください。

#### 重要

現在、パスワード vault は **PKCS12** キーストアタイプをサポートしていません。パスワード vault は、**JCEKS** キーストアタイプのみをサポートします。

使用しているキーストアアルゴリズムに応じて、次のいずれかの **キーサイズ** 値を指定する必要があります。

- AES を使用している場合は、**-keysize 128** を指定します。
- DES を使用している場合は、**-keysize 56** を指定します。
- DESede を使用している場合は、**-keysize 168** を指定します。



## 7.5. APACHE TOMCAT のパスワード VAULT の初期化

**tomcat-vault.sh** スクリプトを使用して、Apache Tomcat のパスワード vault を初期化できます。**tomcat-vault.sh** スクリプトは、パスワード vault を初期化する次のメカニズムのいずれかをサポートします。

- インタラクティブなセットアップ
- 非対話型セットアップ



### 注記

パスワード vault のインストール方法に応じて、**tomcat-vault** スクリプトの場所は異なります。

- アーカイブファイルからパスワードボールドをインストールした場合、**tomcat-vault.sh** スクリプトは **JWS\_HOME/tomcat/bin** ディレクトリーにあります。
- YUM パッケージマネージャーを使用してパスワードボールドをインストールした場合、**tomcat-vault.sh** スクリプトは **/opt/rh/jws5/root/usr/bin** ディレクトリーにあります。

### 7.5.1. Apache Tomcat のパスワード vault の対話的な初期化

Tomcat のパスワード vault を対話的に初期化できます。この場合、**tomcat-vault.sh** スクリプトは、スクリプトの実行中に値を入力するように求めます。

#### 手順

1. **tomcat-vault.sh** スクリプトが含まれるディレクトリーに移動します。
  - アーカイブファイルからパスワードボールドをインストールした場合は、**JWS\_HOME/tomcat/bin** ディレクトリーに移動します。
  - RPM パッケージからパスワードボールドをインストールした場合は、**/opt/rh/jws5/root/usr/bin** ディレクトリーに移動します。
2. **tomcat-vault.sh** スクリプトを実行します。

```
$ ./tomcat-vault.sh
```

3. 画面上の指示に従ってください。  
以下に例を示します。

```
WARNING JBOSS_HOME may be pointing to a different installation - unpredictable results
may occur.
```

```
=====
```

```
JBoss Vault
```

```
JBOSS_HOME: JWS_HOME/tomcat
```

```
JAVA: java
```

```

=====

*****
**** JBoss Vault *****
*****

Please enter a Digit::
0: Start Interactive Session
1: Remove Interactive Session
2: Exit

0

Starting an interactive session
Enter directory to store encrypted files: JWS_HOME/tomcat/
Enter Keystore URL: JWS_HOME/tomcat/vault.keystore
Enter Keystore password: <vault_password>
Enter Keystore password again: <vault_password>
Values match
Enter 8 character salt: 1234abcd
Enter iteration count as a number (Eg: 44): 120
Enter Keystore Alias: my_vault
Initializing Vault
Jun 16, 2018 10:24:27 AM org.apache.tomcat.vault.security.vault.PicketBoxSecurityVault init
INFO: PBOX000361: Default Security Vault Implementation Initialized and Ready
Vault Configuration in tomcat properties file:
*****
...
KEYSTORE_URL=JWS_HOME/tomcat/vault.keystore
KEYSTORE_PASSWORD=MASK-3CuP21KMHn7G6iH/A3YpM/
KEYSTORE_ALIAS=my_vault
SALT=1234abcd
ITERATION_COUNT=120
ENC_FILE_DIR=JWS_HOME/tomcat/
...
*****

Vault is initialized and ready for use
Handshake with Vault complete
Please enter a Digit::
0: Store a secured attribute
1: Check whether a secured attribute exists
2: Exit

2

```

前の例で、指定された設定を環境に適した値に置き換えます。

4. Tomcat プロパティファイルの出力に注意してください。この情報は、パスワード vault を使用するように Tomcat を設定するときが必要です。

## 7.5.2. 非対話型セットアップを使用した Apache Tomcat のパスワード vault の初期化

非対話型セットアップを使用して、Tomcat のパスワード vault を初期化できます。この状況では、スクリプトを実行するときに **tomcat-vault.sh** スクリプトに引数として必要な入力を提供する必要があります。

## 手順

1. **tomcat-vault.sh** スクリプトが含まれるディレクトリーに移動します。
  - アーカイブファイルからパスワードボールドをインストールした場合は、**JWS\_HOME/tomcat/bin** ディレクトリーに移動します。
  - RPM パッケージからパスワードボールドをインストールした場合は、**/opt/rh/jws5/root/usr/bin** ディレクトリーに移動します。
2. **tomcat-vault.sh** スクリプトを実行し、必要な引数を指定します。以下に例を示します。

```
$ ./tomcat-vault.sh \
--keystore JWS_HOME/tomcat/vault.keystore \
--keystore-password <vault_password> \
--alias my_vault \
--enc-dir JWS_HOME/tomcat/ \
--iteration 120 \
--salt 1234abcd \
--generate-config JWS_HOME/tomcat/conf/vault.properties
```

前の例で、指定された設定を環境に適した値に置き換えます。



### 注記

-g、--generate-config オプションを指定すると、**tomcat-vault.sh** スクリプトは、指定されたプロパティーを含む **vault.properties** ファイルも作成します。

## 7.6. パスワード VAULT を使用するための TOMCAT の設定

**vault.properties** ファイルの設定を更新することで、Apache Tomcat がパスワード vault を使用するよう設定できます。

### 前提条件

- Tomcat のパスワード vault を初期化しました。

### 手順

1. **JWS\_HOME/tomcat/conf/** ディレクトリーに移動します。
2. **vault.properties** という名前のファイルを作成します。
3. **vault.properties** ファイルに、Tomcat のパスワード vault を初期化したときに指定したボールド設定プロパティーを入力します。以下に例を示します。

```
KEYSTORE_URL=JWS_HOME/tomcat/vault.keystore
KEYSTORE_PASSWORD=MASK-3CuP21KMHn7G6iH/A3YpM/
KEYSTORE_ALIAS=my_vault
SALT=1234abcd
ITERATION_COUNT=120
ENC_FILE_DIR=JWS_HOME/tomcat/
```



## 注記

前述の例は [Initializing password vault for Apache Tomcat interactively](#) の Vault 設定の例に基づいています。

**KEYSTORE\_PASSWORD** 設定については、パスワード vault を初期化したときに生成されたマスクされた値を使用していることを確認してください。

## 7.7. 外部パスワード VAULT の設定

**JWS\_HOME/tomcat/conf/** ディレクトリーの外に、パスワード vault 用の **vault.properties** ファイルを格納できます。すでに **CATALINA\_BASE/conf/** ディレクトリーを設定している場合は、**vault.properties** ファイルを **CATALINA\_BASE/conf/** ディレクトリーに格納することができます。

**CATALINA\_BASE** ディレクトリーの設定の詳細については、Apache Tomcat Web サイトの [Running The Apache Tomcat 9.0 Servlet/JSP Container](#) の Advanced Configuration - Multiple Tomcat Instances セクションを参照してください。



## 注記

**CATALINA\_BASE** のデフォルトのロケーションは **JWS\_HOME/tomcat/** です。これは、**CATALINA\_HOME** ディレクトリーとも呼ばれます。

### 関連情報

- [Apache Tomcat 9: Introduction - Directories and Files](#)
- [Running The Apache Tomcat 9.0 Servlet/JSP Container: "Advanced Configuration - Multiple Tomcat Instances"](#)

## 7.8. パスワード VAULT への SENSITIVE 文字列の格納

**tomcat-vault.sh** スクリプトを使用して、機密性の高い文字列をパスワード vault に保存できます。**tomcat-vault.sh** スクリプトは、対話モードまたは非対話モードで実行できます。

機密性の高い文字列をパスワード vault に追加する場合は、文字列の名前を指定する必要があります。この場合、文字列の名前は **attribute name** と呼ばれ、文字列自体は **secured attribute** と呼ばれます。

### 手順

1. **tomcat-vault.sh** スクリプトが含まれるディレクトリーに移動します。
  - アーカイブファイルからパスワードボールドをインストールした場合は、**JWS\_HOME/tomcat/bin** ディレクトリーに移動します。
  - RPM パッケージからパスワードボールドをインストールした場合は、**/opt/rh/jws5/root/usr/bin** ディレクトリーに移動します。
2. 非対話モードで **tomcat-vault.sh** スクリプトを使用するには、次のコマンドを入力します。

```
$ ./tomcat-vault.sh \
--keystore JWS_HOME/tomcat/vault.keystore \
--keystore-password <vault_password> \
--alias my_vault \
```

```
--enc-dir JWS_HOME/tomcat \
--iteration 120 \
--salt 1234abcd \
--vault-block my_block \
--attribute manager_password \
--sec-attr P@SSW0#D
```

### 注記

前述の例は [Initializing password vault for Apache Tomcat interactively](#) の Vault 設定の例に基づいています。前の例では、属性名 **manager\_password** を使用して機密文字列 **P@SSW0#D** を格納しています。

**tomcat-vault.sh** スクリプトを実行すると、必要に応じて、パスワードを保存する vault ブロックを指定できます。ブロックを指定しない場合、**tomcat-vault.sh** スクリプトによって自動的にブロックが作成されます。前の例では、**my\_block** という名前の vault ブロックを指定しています。

## 7.9. TOMCAT 設定で保存された機密文字列の使用

機密性の高い文字列をパスワード vault に保存する場合、設定ファイルで実際の文字列を指定するのではなく、属性名を参照できます。セキュリティーで保護された文字列を文字列の属性名に置き換えることで、Tomcat 設定ファイルにパスワードへの参照のみが含まれるようにすることができます。この場合、実際のパスワードはパスワード vault にのみ保存されます。

### 手順

1. 機密文字列を含む Tomcat 設定ファイルを開きます。
2. 機密文字列を文字列の属性名に置き換え、次の形式で属性名を入力してください:  
**`\${VAULT::block\_name::attribute\_name::}`**  
以下に例を示します。

セキュリティーで保護された文字列 **P@SSW0#D** の次のファイルエントリーの例を考えてみましょう。

```
<user username="manager" password="P@SSW0#D" roles="manager-gui"/>
```

保護された文字列 **P@SSW0#D** に属性名 **manager\_password** がある場合、保護された文字列を次の値に置き換えます。

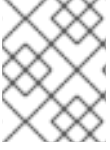
```
<user username="manager" password="`${VAULT::my_block::manager_password::}`"
roles="manager-gui"/>
```

### 注記

前の例は、[機密性の高い文字列をパスワード vault に保存する](#) 設定例に基づいています。前の例では、機密性の高い文字列 **P@SSW0#D** を、**my\_block** というブロックにある属性名 **manager\_password** に置き換えています。

## 第8章 SSI フィルターの設定

JBoss Web Server のフィルターベースの Server Side Includes (SSI) サポートを設定して、既存の HTML ページでのコンテンツの動的生成を有効にすることができます。



### 注記

以前のバージョンのように SSI フィルターを設定しようとする、SSI ディレクティブは機能しません。

### 手順

1. **conf/web.xml** ファイルを開きます。
2. **web.xml** ファイルで、次のブロックのコメントを外します。

```
<mime-mapping>  
  <extension>shtml</extension>  
  <mime-type>text/x-server-parsed-html</mime-type>  
</mime-mapping>
```

## 第9章 RED HAT JBOSS WEB SERVER の FIPS の設定

JBoss Web Server が Red Hat Enterprise Linux 8 ホストにインストールされている場合は、Federal Information Processing Standards (FIPS) に準拠するよう JBoss Web Server を設定できます。Red Hat Enterprise Linux ホストで FIPS を有効にすると、JBoss Web Server は自動的に FIPS モードで動作できるようになります。



### 注記

FIPS は、JBoss Web Server の **tomcat-vault** コンポーネントによって提供されるパスワードベースの暗号化機能をサポートしません。JBoss Web Server ホストでパスワードベースの暗号化を使用する場合は、FIPS が無効になっていることを確認する必要があります。パスワードベースの暗号化および **tomcat-vault** の詳細は、[Red Hat JBoss Web Server の vault](#) を参照してください。

### 9.1. FIPS の概要

FIPS (Federal Information Processing Standards) は、コンピューターシステムやネットワーク間のセキュリティおよび相互運用性を強化するためのガイドラインと要件を提供します。FIPS 140-2 および 140-3 シリーズは、ハードウェアおよびソフトウェアの両レベルで暗号化モジュールに適用されます。アメリカ国立標準技術研究所は、進行中の暗号モジュールと承認済みの暗号モジュールの両方の検索可能なリストとともに [暗号化モジュール検証プロウラム](#) を実装しています。

Red Hat Enterprise Linux は、FIPS 140-2 コンプライアンスをシステム全体で有効にする統合フレームワークを提供します。FIPS モードで操作する場合、暗号化ライブラリーを使用するソフトウェアパッケージはグローバルポリシーに従って自己設定されます。

#### 関連情報

- [Government Standards](#) (Red Hat カスタマーポータル)
- [Security Requirements for Cryptographic Modules](#) (National Institute of Standards and Technology (NIST) の Web サイト)

### 9.2. RHEL 8 での JBOSS WEB SERVER の FIPS の設定

システムのインストール時に、Red Hat Enterprise Linux 8 ホストで FIPS コンプライアンスを有効にできます。または、システムのインストールが完了した後に FIPS モードに切り替えることができます。

#### 手順

- FIPS モードを有効にするには、以下のいずれかの手順を実行します。
  - システムのインストール中に FIPS を有効にする場合は、[Security Hardening: Installing the system with FIPS mode enabled](#) の手順に従ってください。
  - システムのインストール後に FIPS モードに切り替える場合は、[Security Hardening: Switching the system to FIPS mode](#) の手順に従ってください。

#### 検証

- 以下のコマンドを入力します。

```
fips-mode-setup --check
```

FIPS が有効にされている場合、これは以下の出力を表示します。

```
█ FIPS mode is enabled.
```



## 付録A JAVA IPV4 および IPV6 プロパティ

Java プロパティを使用して、IPv4 および IPv6 アドレスを設定できます。その後、これらのプロパティを Tomcat にエクスポートし、アドレス値を使用して Tomcat バインディングを指定できます。

### A.1. JAVA IPV4 および IPV6 プロパティの概要

Java には、IPv4 および IPv6 アドレスの設定に使用できる 2 つのプロパティが用意されています。

#### java.net.preferIPv4Stack (default: false)

IPv6 が利用できる場合は、基礎となるネイティブソケットがデフォルトで IPv6 ソケットになります。このソケットにより、アプリケーションは IPv4 および IPv6 ホストに接続し、接続を受け入れることができます。アプリケーションが IPv4 ソケットのみを使用する場合は、このプロパティを **true** に設定します。ただし、IPv4 ソケットのみを使用するアプリケーションは、IPv6 のみのホストと通信できません。

#### java.net.preferIPv6Addresses (default: false)

ホストに IPv4 アドレスと IPv6 アドレスの両方があり、IPv6 が利用できる場合には、デフォルトの動作では IPv6 上で IPv4 アドレスを使用します。これにより、後方互換性を維持できます。アプリケーションが 192.168.1.1 などの IPv4 アドレス表現に依存している場合は、このプロパティを **true** に設定して設定を変更し、可能であれば IPv4 ではなく IPv6 アドレスを使用します。

### A.2. JAVA IPV4 および IPV6 プロパティの TOMCAT へのエクスポート

**JWS\_HOME/tomcat/bin/setenv.\*** ファイルで **CATALINA\_OPTS** を設定することにより、Java IPv4 および IPv6 プロパティを Tomcat にエクスポートできます。Red Hat Enterprise Linux では、**setenv** ファイルの拡張子は、**.sh** です。Microsoft Windows では、**setenv** ファイルの拡張子は **.bat** です。

#### 手順

1. **JWS\_HOME/tomcat/bin/setenv.\*** ファイルが存在しない場合は、ファイルを作成します。



#### 注記

Red Hat Enterprise Linux を使用している場合は、**setenv.sh** ファイルを作成します。Microsoft Windows を使用している場合は、**setenv.bat** ファイルを作成します。

2. Java IPv4 および IPv6 プロパティを Tomcat にエクスポートするには、次のいずれかの手順を実行します。

- Red Hat Enterprise Linux を使用している場合は、次のコマンドを入力します。

```
export "CATALINA_OPTS=-Djava.net.preferIPv4Stack=YOUR_VALUE -
Djava.net.preferIPv6Addresses=YOUR_VALUE"
```

- Microsoft Windows を使用している場合は、次のコマンドを入力します。

```
set "CATALINA_OPTS=-Djava.net.preferIPv4Stack=YOUR_VALUE -
Djava.net.preferIPv6Addresses=YOUR_VALUE"
```

### A.3. TOMCAT バインディングの設定

IPv6 アドレスを指定することにより、**JWS\_HOME/tomcat/conf/server.xml** ファイルで Tomcat バインディングを設定できます。

## 手順

1. **JWS\_HOME/tomcat/conf/server.xml** ファイルを開きます。
2. Tomcat バインディングアドレスを指定するには、次の詳細を入力します。

```
<Server ... address="TOMCAT_BINDING_ADDRESS">
```

3. HTTP コネクターアドレスを指定するには、次の詳細を入力します。

```
<Connector protocol="HTTP/1.1" ... address="HTTP_CONNECTOR_ADDRESS">
```

4. AJP コネクターアドレスを指定するには、次の詳細を入力します。

```
<Connector protocol="AJP/1.3" ... address="AJP_CONNECTOR_ADDRESS">
```



### 注記

**TOMCAT\_BINDING\_ADDRESS**、**HTTP\_CONNECTOR\_ADDRESS**、および **AJP\_CONNECTOR\_ADDRESS** を正しい IPv6 アドレスに置き換えてください。