



## Red Hat JBoss Web Server 5.5

### インストールガイド

Red Hat JBoss Web Server 5.5 のインストールおよび設定



# Red Hat JBoss Web Server 5.5 インストールガイド

---

## Red Hat JBoss Web Server 5.5 のインストールおよび設定

Enter your first name here. Enter your surname here.

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

Enter your email address here.

## 法律上の通知

Copyright © 2021 | You need to change the HOLDER entity in the en-US/Installation\_Guide.ent file |.

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

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

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

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

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

Linux<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

本書には、Red Hat JBoss Web Server のインストールおよび設定に関連する情報が記載されています。

## 目次

多様性を受け入れるオープンソースの強化 .....	4
<b>第1章 はじめに .....</b>	<b>5</b>
1.1. RED HAT JBOSS WEB SERVER	5
1.1.1. コンポーネントの完全リスト	5
1.2. サポート対象オペレーティングシステムおよび設定	6
1.3. RED HAT JBOSS WEB SERVER をインストールする方法	7
1.4. コンポーネントドキュメントバンドル	7
<b>第2章 RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER のインストール .....</b>	<b>8</b>
2.1. 前提条件	8
2.1.1. YUM パッケージマネージャーを使用した Java Development Kit (JDK) のインストール	8
2.1.2. 圧縮アーカイブ (.zip、.tar など) からの JDK のインストール	9
2.1.3. Red Hat Enterprise Linux パッケージの要件	9
2.2. JBOSS WEB SERVER (ZIP) のインストールおよび管理	10
2.2.1. JBoss Web Server のダウンロードおよび拡張	10
2.2.2. Red Hat Enterprise Linux での JBoss Web Server の管理	11
2.2.2.1. Red Hat Enterprise Linux での .zip インストールにシステムデーモンを使用した JBoss Web Server の管理	11
2.2.2.1.1. systemd による JBoss Web Server の設定および使用	11
2.2.2.2. コマンドラインでの JBoss Web Server の管理	12
2.2.2.2.1. JBoss Web Server インストールの設定	12
2.2.2.2.2. JBoss Web Server の起動	13
2.2.2.2.3. JBoss Web Server の停止	13
2.2.3. RPM インストール	13
2.3.1. Red Hat Enterprise Linux へのサブスクリプションの割り当て	14
2.3.2. 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	14
2.3.3. JBoss Web Server の起動	15
2.3.4. JBoss Web Server の停止	16
2.3.5. 起動時に開始する JBoss Web Server サービスの設定	16
2.4. SELINUX ポリシー	17
2.4.1. SELinux ポリシー情報	17
2.4.2. RPM インストール用の SELinux ポリシー	17
2.4.3. アーカイブインストールの SELinux ポリシー	17
<b>第3章 MICROSOFT WINDOWS への JBOSS WEB SERVER のインストール .....</b>	<b>19</b>
3.1. JAVA DEVELOPMENT KIT (JDK) のインストール	19
3.2. JBOSS WEB SERVER のダウンロードおよび拡張	19
3.3. JBOSS WEB SERVER インストールの設定	20
3.4. JBOSS WEB SERVER の起動	21
3.5. JBOSS WEB SERVER の停止	21
<b>第4章 RED HAT JBOSS WEB SERVER の HIBERNATE の設定 .....</b>	<b>23</b>
4.1. HIBERNATE ORM のインストール	23
4.2. JDBC 接続プールの設定	23
4.3. HIBERNATE 接続プロパティの設定	23
4.4. JDBC データソースの追加	24
<b>第5章 RED HAT JBOSS WEB SERVER の HTTP/2 の有効化 .....</b>	<b>25</b>
要件	25
手順	25
次のステップ	26
その他のリソース	27

---

<b>第6章 RED HAT JBOSS WEB SERVER の VAULT</b> .....	<b>28</b>
6.1. RED HAT JBOSS WEB SERVER 5.5 のパスワード VAULT	28
6.2. ZIP アーカイブからの JBOSS WEB SERVER パスワード VAULT のインストール	28
6.3. YUM パッケージマネージャーを使用した RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER パスワード VAULT のインストール	28
6.4. JBOSS WEB SERVER でのパスワード VAULT の有効化	28
6.5. JBOSS WEB SERVER での JAVA キーストアの作成	28
6.6. パスワード VAULT の初期化	29
6.6.1. Apache Tomcat のパスワード vault の対話的な初期化	29
6.6.2. Apache Tomcat の Vault の非対話的な初期化 (サイレント設定)	30
6.7. パスワード VAULT を使用するための TOMCAT の設定	31
6.8. 外部パスワード VAULT の設定	31
6.9. パスワード VAULT への SENSITIVE 文字列の格納	32
6.10. TOMCAT 設定で保存された機密文字列の使用	32
<b>第7章 SSI フィルターの設定</b> .....	<b>34</b>
7.1. SSI フィルターの設定	34
<b>付録A JAVA IPV4/IPV6 プロパティ</b> .....	<b>35</b>
Java プロパティの設定	35
Tomcat バインディングの設定	35



## 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、[弊社](#) の CTO、Chris Wright の [メッセージ](#) を参照してください。



## 第1章 はじめに

このインストールガイドには、サポートされるオペレーティングシステム上で JBoss Web Server から Tomcat サーバーをインストール、マイナーアップグレード、および基本設定する手順が含まれています。Apache HTTP Server のインストールおよび設定手順は、[JBoss Core Services ドキュメンテーション](#)を参照してください。

### 1.1. RED HAT JBOSS WEB SERVER

Red Hat JBoss Web Server は、Java Web アプリケーションをホストするための完全統合および認定されたコンポーネントセットです。以下のコンポーネントで構成されます。

- アプリケーションサーバー (Apache Tomcat サブレットコンテナ)
- Tomcat Native Library

#### 1.1.1. コンポーネントの完全リスト

Red Hat JBoss Web Server には以下のコンポーネントが含まれます。

コンポーネント	バージョン
Apache CXF	3.3.5
Apache Tomcat 9	9.0.43
ECJ	4.12.0
Hibernate	5.3.20.Final
JBoss Logging	3.4.1.Final
libapr	1.6.3
mod_cluster	1.4.3.Final
OpenSSL	1.1.1g
Tomcat-Native	1.2.26
Tomcat-Vault	1.1.8.Final



## 重要

Apache Tomcat は、RHEL 7 の一部として RHEL プラットフォームのサブスクリプションで提供されていますが、RHEL 8 には**提供されません**。今後のリリースでは、Tomcat は Middleware Runtimes サブスクリプションの一部として取得できます。RHEL が提供する Tomcat と、Runtimes JWS サブスクリプションで提供される Tomcat にはいくつかの違いがあります。RHEL 7 には Tomcat 7 が同梱されています。JWS バージョン 3.1 は Tomcat 7 および 8 を提供し、JWS バージョン 5.x は Tomcat 9 を提供します。また、両方のプロジェクトは RPM パッケージを提供しますが、JWS のみが .ZIP アーカイブを提供します。



## 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。

主なコンポーネントの説明は次のとおりです。

- **Apache tomcat:** Java Servlet Specification に準拠するサーブレットコンテナ。JBoss Web Server には Apache Tomcat 9 が含まれています。
- **Apache tomcat ネイティブライブラリー:** Tomcat ライブラリーでは、Tomcat のスケーラビリティ、パフォーマンス、ネイティブサーバー技術との統合が改善されます。
- **Tomcat-vault:** JBoss Web Server によって使用されるパスワードおよび他の機密情報をセキュアに保存するために使用される JBoss Web Server の拡張機能。
- **mod\_cluster** ライブラリー: Apache Tomcat と Apache HTTP Server の **mod\_proxy\_cluster** モジュール間の通信を可能にするライブラリー。これにより、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、**mod\_jk** および **mod\_proxy** の代替ロードバランサーのインストールおよび設定の詳細は、『[HTTP Connectors and Load Balancing Guide](#)』を参照してください。
- **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、以下にアクセスできます。
  - 高度な IO 機能: sendfile、epoll、OpenSSL など。
  - オペレーティングシステムレベルの機能: 乱数の生成やシステムステータスなど。
  - ネイティブプロセス処理 - 共有メモリー、NT パイプ、Unix ソケットなど
- **OpenSSL:** Secure Sockets Layer (SSL) プロトコルおよび TLS (Transport Layer Security) プロトコルを実装するソフトウェアライブラリー。基本的な暗号化ライブラリーが含まれています。



## 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レプリケーションのサポートが必要な場合、Red Hat は Red Hat JBoss Enterprise Application Platform (JBoss EAP) を使用することを推奨します。

## 1.2. サポート対象オペレーティングシステムおよび設定

Red Hat JBoss Web Server は、以下のオペレーティングシステムおよび設定をサポートします。

### 1.3. RED HAT JBOSS WEB SERVER をインストールする方法

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すると、サポートされる Red Hat Enterprise Linux および Microsoft Windows システムに 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-5.5.0-application-server.zip**
  - Tomcat 9
  - mod\_cluster
  - tomcat-vault
- **jws-5.5.0-application-server-<platform>-<architecture>.zip**
  - プラットフォーム固有のユーティリティ

### 1.4. 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-5.5.0.zip** は Red Hat カスタマーポータルで入手できます。また、以下のコンポーネントの補足ドキュメントが含まれています。

- tomcat
- tomcat-native
- tomcat-vault

## 第2章 RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER のインストール

JBoss Web Server は、以下のいずれかの方法を使用して Red Hat Enterprise Linux にインストールできます。

- [アーカイブファイル](#)
- [RPM パッケージ](#)

選択した方法に関係なく、最初に [サポートされる Java Development Kit \(JDK\) をインストール](#) する必要があります。

Red Hat Enterprise Linux-7 および Red Hat Enterprise Linux-8 の前提条件は、[Red Hat Enterprise Linux パッケージの要件](#) を参照してください。

### 2.1. 前提条件

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

JBoss Web Server をインストールする前に、サポートされる Java Development Kit (JDK) を最初にインストールする必要があります。

サポートされる JDK の完全リストは、「[サポートされるオペレーティングシステムおよび設定](#)」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
- **IBM:**
  - rhel-7-server-supplementary-rpms
  - rhel-8-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。

1. 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
```

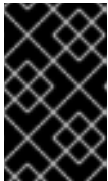
<VENDOR> を **ibm** または **openjdk** に置き換えます。

2. 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一覧を返します。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 (.zip、.tar など) からの JDK のインストール

JBoss Web Server をインストールする前に、サポートされる Java Development Kit (JDK) を最初にインストールする必要があります。

サポートされる JDK の完全な一覧は、本ガイドの [1.2](#) に記載されています。

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。

JDK が圧縮アーカイブからインストールされている場合は、JBoss Web Server を実行する前に Tomcat の **JAVA\_HOME** 環境変数を設定します。

Tomcat の **bin** ディレクトリー (**JWS\_HOME/tomcat/bin**) では、**setenv.sh** という名前のファイルを作成し、**JAVA\_HOME** パス定義を挿入します。

以下は例になります。

```
$ 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、以下の前提条件を満たしていることを確認してください。

- [サポートされる JDK がインストールされている](#)。
- さらに、OpenSSL または APR を必要とする **RHEL 8** ユーザーは、オペレーティングシステムからインストールする必要があります。OpenSSL および APR をインストールするには、以下のコマンドを実行します。

```
# yum install openssl
```

```
# 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 セキュリティモデルではなく、基礎となる NSS セキュリティモデルを使用します。
  - root ユーザーとして以下のコマンドを実行し、**tomcatjss** を削除します。

```
# yum remove tomcatjss
```

### 重要

- RHEL 7 では、JWS は Red Hat JBoss Core Services からの OpenSSL および APR を使用しますが、RHEL 8 OpenSSL と APR はオペレーティングシステムから使用されます。
- RHEL 8 zip パッケージには、オペレーティングシステムからインストールする必要のある OpenSSL および APR が含まれていません。

## 2.2. JBOSS WEB SERVER (ZIP) のインストールおよび管理

アーカイブファイルから JBoss Web Server をインストールできます。アーカイブからインストールする場合、RPM パッケージのインストールと比較すると、製品の管理方法が異なります。たとえば、起動時にシステムデーモンを使用し、コマンドラインから JBoss Web Server を管理できます。まず、アーカイブファイルをダウンロードして展開します。

### 2.2.1. JBoss Web Server のダウンロードおよび拡張

このインストールでは、[Red Hat カスタマーポータル](#) にアクセスして、JBoss Web Server の正しいバージョンを見つける必要があります。

#### 要件

- JBoss Web Server をインストールする前に、すべての [前提条件を満たし](#) ていることを確認してください。

#### 手順

JBoss Web Server をインストールするには、インストールアーカイブファイルをダウンロードおよび展開します。

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. **ダウンロード** 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が正しく選択されていることを確認します。
  - 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展開します。

例を以下に示します。

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

アーカイブを抽出して作成されたディレクトリーは、JBoss Web Server の最上位ディレクトリーになります。これは **JWS\_HOME** と呼ばれます。

## 2.2.2. Red Hat Enterprise Linux での JBoss Web Server の管理

Red Hat Enterprise Linux で Red Hat JBoss Web Server を実行し、管理する方法は 2 つあります。

- システムデーモンの使用
- コマンドライン

JBoss Web Server を管理するのに推奨される方法はシステムデーモンを使用することです。

### 2.2.2.1. Red Hat Enterprise Linux での .zip インストールにシステムデーモンを使用した JBoss Web Server の管理

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 8 および Red Hat Enterprise Linux 7 のデフォルトのシステムデーモンは systemd です。

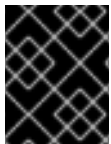


#### 注記

実行中のシステムデーモンを確認するには、**ps -p 1 -o comm=** を実行します。

- systemd の場合:

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



#### 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。

#### 2.2.2.1.1. systemd による JBoss Web Server の設定および使用

##### systemd 用 JBoss Web Server の設定

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
```

##### systemd を使用した JBoss Web Server の制御

systemd コマンドは、root ユーザーのみが実行できます。



- JBoss Web Server サービスが、systemd を使用して起動時に起動するようにするには、以下を行います。

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

- systemd を使用して JBoss Web Server を起動するには、以下を行います。

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



### 注記

**SECURITY\_MANAGER** 変数は RHEL zip インストールに基づく設定で非推奨となり、以下のコメントが追加されます。

```
# 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、以下を実行します。

```
# 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、以下を実行します (任意のユーザーが **status** 操作を実行できます)。

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

RHEL 7 で systemd を使用方法は、『[RHEL 7 System Administrator's Guide: Managing System Services](#)』を参照してください。

RHEL 8 で systemd を使用方法は、『[RHEL 8 Configuring Basic System Settings: Managing system services with systemctl](#)』を参照してください。

## 2.2.2.2. コマンドラインでの JBoss Web Server の管理

### 2.2.2.2.1. JBoss Web Server インストールの設定



### 注記

以下の設定手順は、『[Red Hat Enterprise Linux での .zip インストール用のシステムデーモンを使用した JBoss Web Server の管理](#)』で説明されている **.postinstall.sysv** スクリプトおよび **.postinstall.systemd** スクリプトにより実行されます。

JBoss Web Server を実行する前に一部の設定が必要になります。本セクションには、以下の設定手順が含まれています。

- [JAVA\\_HOME 環境変数の設定](#)
- [簡単でセキュアなユーザー管理のための tomcat ユーザーの作成: Tomcat ユーザーの作成](#)



- **tomcat** ディレクトリーの所有権を **tomcat** ユーザーに移動して、**tomcat** ユーザーに JBoss Web Server へのアクセス権限を付与します。

### JAVA\_HOME 環境変数の設定

JBoss Web Server を実行する前に、Tomcat の **JAVA\_HOME** 環境変数を設定する必要があります。

Tomcat の **bin** ディレクトリー (**JWS\_HOME/tomcat/bin**) では、**setenv.sh** という名前のファイルを作成し、**JAVA\_HOME** パス定義を挿入します。

例: **JAVA\_HOME=/usr/lib/jvm/jre-1.8.0-openjdk.x86\_64**

### Tomcat ユーザーの作成

以下の手順に従って、**tomcat** ユーザーと親グループを作成します。

1. root ユーザーとしてシェルプロンプトで、**JWS\_HOME** ディレクトリーに移動します。
2. 以下のコマンドを実行して **tomcat** ユーザーグループを作成します。

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

3. 以下のコマンドを実行して、**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
```

**tomcat** ディレクトリーの所有権を **tomcat** ユーザーに移動します。

1. **JWS\_HOME** から次のコマンドを実行して、Tomcat ディレクトリーの所有権をユーザーに割り当てて、**tomcat** ユーザーが Tomcat サービスを実行できるようにします。

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

**ls -l** を使用して、**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できます。

2. **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。例を以下に示します。

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

#### 2.2.2.2.2. JBoss Web Server の起動

**tomcat** ユーザーとして以下のコマンドを実行します。

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

#### 2.2.2.2.3. JBoss Web Server の停止

Tomcat を停止するには、**tomcat** ユーザーとして以下のコマンドを実行します。

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

## 2.3. RPM インストール

RPM パッケージから JBoss Web Server をインストールすると、Tomcat がサービスとしてインストールされ、そのリソースを絶対パスにインストールします。RPM インストールオプションは、Red Hat Enterprise Linux 7、および Red Hat Enterprise Linux 8 で利用できます。

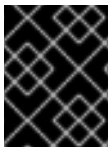
JBoss Web Server の RPM インストールパッケージは、Red Hat Subscription Management から入手できます。

### 2.3.1. Red Hat Enterprise Linux へのサブスクリプションの割り当て

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、システムを Red Hat Subscription Management に登録し、該当する Content Delivery Network (CDN) リポジトリをサブスクライブする必要があります。

Red Hat Enterprise Linux の登録に関する情報は、以下の手順を参照してください。

- [Red Hat Enterprise Linux 7 のサブスクリプションマネージャー](#)
- [Red Hat Enterprise Linux 8 のサブスクリプションマネージャー](#)



#### 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 Manager](#) にログインします。
2. **Systems** タブをクリックします。
3. システムの **Name** をクリックして、サブスクリプションを追加します。
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選択してから、**Attach Subscriptions** をクリックします。



#### 注記

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には、以下を行います。

1. <https://access.redhat.com/management/subscriptions> にログインします。
2. **Subscription Name** をクリックします。
3. **Products Provided** には以下が必要です。
  - JBoss Enterprise Web Server
  - Red Hat JBoss Core Services

### 2.3.2. 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

#### 要件

- Java Development Kit (JDK) をインストールします
- tomcatjss パッケージが削除されていることを確認します

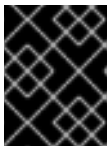
## 手順

1. コマンドラインで、**subscription-manager** を使用して、お使い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します。

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

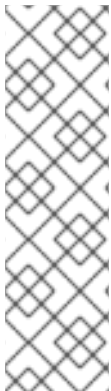
- Red Hat Enterprise Linux 7 の場合
    - jws-5-for-rhel-7-server-rpms
    - jb-coreservices-1-for-rhel-7-server-rpms
  - Red Hat Enterprise Linux 8 の場合
    - jws-5-for-rhel-8-x86\_64-rpms
2. root ユーザーとして以下のコマンドを実行し、JBoss Web Server をインストールします。

```
# yum groupinstall jws5
```



### 重要

RPM ディストリビューションの場合、JWS\_HOME フォルダは `/opt/rh/jws5/root/usr/share` になります。



### 注記

- 推奨されませんが、`group install` を使用する代わりに、各パッケージと依存関係を個別にインストールできます。
- RHEL 8 システムを除く JBoss Web Server のインストールには、上記の Red Hat JBoss Core Services リポジトリが必要です。
- 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る。この機能の詳細は、『Packaging Guide』の「[Using Software Collections over NFS](#)」を参照してください。

## 2.3.3. JBoss Web Server の起動

この手順では、JBoss Web Server を起動する方法を説明します。



### 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 ユーザーとしてシェルプロンプトで、Tomcat サービスを起動します。

- Red Hat Enterprise Linux 7 または 8 の場合

```
# 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。

- Tomcat が稼働していることを確認するには、service **status** コマンドの出力を確認する必要があります。これは、任意のユーザーとして実行できます。

- Red Hat Enterprise Linux 7 または 8 の場合

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

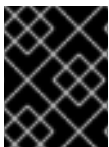


#### 注記

RHEL 8 での HTTPD のインストールおよび設定に関する完全な手順は、[このリンク](#)を参照してください。

### 2.3.4. JBoss Web Server の停止

この手順では、JBoss Web Server を停止する方法を示しています。



#### 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 ユーザーとしてシェルプロンプトで、Tomcat サービスを停止します。
  - Red Hat Enterprise Linux 7 または 8 の場合

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

- Tomcat が実行しなくなったことを確認するには、service **status** コマンドの出力を確認する必要があります。これは、任意のユーザーとして実行できます。

- Red Hat Enterprise Linux 7 または 8 の場合

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



#### 注記

RHEL 8 での HTTPD のインストールおよび設定に関する完全な手順は、[このリンク](#)を参照してください。

### 2.3.5. 起動時に開始する JBoss Web Server サービスの設定

以下のコマンドを使用して、JBoss Web Server サービスが起動時に開始できるようにします。



### 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

- Red Hat Enterprise Linux のバージョンに応じて、以下のコマンドのいずれかを入力します。
  - Red Hat Enterprise Linux 7 または 8 の場合

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

## 2.4. SELINUX ポリシー

### 2.4.1. SELinux ポリシー情報

以下の表は、jws5-tomcat-selinux パッケージで提供される SELinux ポリシーを説明します。

表2.1RPM 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
jws5_tomcat	<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。	jws5_tomcat 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するために、tomcat が <b>/var/opt/rh/jws5/lib/tomcat</b> 、 <b>/var/opt/rh/jws5/log/tomcat</b> 、 <b>/var/opt/rh/jws5/cache/tomcat</b> 、 <b>/var/opt/rh/jws5/run/tomcat.pid</b> を書き込みできるようにします。

SELinux およびその他の Red Hat Enterprise Linux セキュリティ情報の使用に関する詳細は『Red Hat Enterprise Linux Security Guide』を参照してください。

### 2.4.2. RPM インストール用の SELinux ポリシー

JBoss Web Server の SELinux ポリシーは、**jws5-tomcat-selinux** パッケージで提供されます。これらのパッケージは JWS チャネルで利用できます。

JBoss Web Server 5.5 の SELinux ポリシーを有効にするには、**jws5-tomcat-selinux** パッケージをインストールします。

### 2.4.3. アーカイブインストールの SELinux ポリシー

本リリースでは、SELinux ポリシーがアーカイブパッケージに提供されます。SELinux セキュリティモデルはカーネルにより適用され、アプリケーションがファイルシステムの場所やポートなどのリソースに限定的なアクセスを行えるようにします。これにより、誤ったプロセス (危険にさらされるか、設

定されていないかのいずれか) が制限され、場合によっては実行できないようにすることが可能になります。

**.postinstall.selinux** ファイルは、`jws -5.5.0-application-server-<platform> -<architecture>.zip` の **tomcat** フォルダに含まれます。必要に応じて、**.postinstall.selinux** スクリプトを実行できます。

アーカイブを使用して SELinux ポリシーをインストールするには、以下を実行します。

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

JBoss Web Server に追加ポートが必要な場合は、**semanage** コマンドを使用して必要なパーミッションを提供し、ポート番号を必要なポートに置き換えます。

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

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するには、以下のように設定します。

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

## 注記

デフォルトでは、提供される SELinux ポリシーはアクティブではなく、Tomcat プロセスが **unconfined\_java\_t** ドメイン内で実行されます。このドメインはプロセスに制限されず、提供される SELinux ポリシーを有効にしなかった場合は、以下のセキュリティー対策を実行することが推奨されます。

- **tomcat** ユーザーのファイルアクセスを JBoss Web Server ランタイムに必要なファイルおよびディレクトリーのみに制限します。
- **root** ユーザーで Tomcat を実行しないでください。

## 第3章 MICROSOFT WINDOWS への JBOSS WEB SERVER のインストール

### 3.1. JAVA DEVELOPMENT KIT (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、最初にサポートされる Java Development Kit (JDK) をインストールする必要があります。

サポートされる構成の一覧は、「[サポートされるオペレーティングシステムおよび設定](#)」を参照してください。



#### 注記

IBM JDK のインストール方法は、<https://www.ibm.com/developerworks/java/jdk/> を参照してください。

Oracle Java Development Kit をインストールするには、以下を実行します。

1. 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。JDK インストールファイルは、Oracle の Web サイト <http://www.oracle.com/technetwork/java/javase/downloads/index.html> からダウンロードできます。
2. 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。
3. インストールウィンドウで指示どおりに続行します。

### 3.2. JBOSS WEB SERVER のダウンロードおよび拡張

JBoss Web Server をインストールするには、インストールアーカイブファイルをダウンロードおよび展開します。

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. **ダウンロード** 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。
  - 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**)
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展開します。

アーカイブを抽出して作成されたディレクトリーは、JBoss Web Server の最上位ディレクトリーになります。これは **JWS\_HOME** と呼ばれます。

### 3.3. JBOSS WEB SERVER インストールの設定

JBoss Web Server を実行する前に一部の設定が必要になります。本セクションには、以下の設定手順が含まれています。

- [環境変数の設定](#)
- [Tomcat サービスのインストール](#)
- [JBoss Web Server サービスのフォルダーパーミッションの設定](#)

#### 環境変数の設定

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. **コントロールパネル** → **システム** と選択します。
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 コネクターが機能するには、サービスを実行するユーザーの **PATH** 環境変数に **JWS\_HOME\bin** を追加する必要があります。このユーザーは、デフォルトで **SYSTEM** です。

#### Tomcat サービスのインストール

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. 以下のコマンドを使用して、Tomcat サービスをインストールします。

```
call service.bat install
```

#### JBoss Web Server サービスのフォルダーパーミッションの設定

以下の手順に従って、サービスの実行に使用するアカウントが **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します。

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。

### 3.4. JBOSS WEB SERVER の起動

コマンドプロンプトまたは Computer Management ツールから JBoss Web Server を起動できます。

#### コマンドプロンプトからの JBoss Web Server の起動

1. 管理者権限でコマンドプロンプトを開きます。
2. Tomcat サービスを起動します。

```
net start tomcat9
```

#### Computer Management Tool からの JBoss Web Server の起動

1. **Start** → **Administrative Tools** → **Services** と選択します。
2. **Services** 一覧で、サービス名 (**Tomcat9**) を右クリックし、**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。これは、検索時に Tomcat ライブラリーよりも優先されます。つまり、これらのサードパーティーライブラリーの名前が Tomcat ネイティブライブラリーで使用されるものと同じ場合は、JBoss Web Server と配布されるライブラリーの代わりにロードされます。

この場合、Tomcat は起動しない可能性があり、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。エラーが発生するのは、**catalina.bat run** の使用した場合のみです。

この動作が発生した場合は、**C:\windows\System32\**ディレクトリーおよびその他の **PATH** ディレクトリーの内容を検査し、JBoss Web Server に同梱される DLL がないことを確認します。特に、**libeay32.dll**、**ssleay32.dll**、**libssl32.dll** を検索します。

### 3.5. JBOSS WEB SERVER の停止

コマンドプロンプトまたは Computer Management ツールを使用して JBoss Web Server を停止できます。

#### コマンドプロンプトからの JBoss Web Server の停止

1. 管理者権限でコマンドプロンプトを開きます。
2. Tomcat サービスを停止します。

```
net stop tomcat9
```

#### Computer Management Tool からの JBoss Web Server の停止

1. Start → Administrative Tools → Services と選択します。
2. Services 一覧で、サービス名 (**Tomcat9**) を右クリックし、**Stop** をクリックします。

## 第4章 RED HAT JBOSS WEB SERVER の HIBERNATE の設定

Hibernate ORM は、JBoss Web Server を JDBC データソースに接続できるようにす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。

### 4.1. HIBERNATE ORM のインストール

以下の手順に従って、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 をインストールします。

#### 前提条件

**Jboss-web-server-5.5.0-maven-repository.zip** としてダウンロード可能な JBoss Web Server Maven リポジトリを使用するようプロジェクトを設定します。

#### 手順

1. JBoss Web Server Maven リポジトリから 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。

#### リファレンス

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

### 4.2. JDBC 接続プールの設定

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。

#### 手順

1. デプロイメントの **/META-INF/context.xml** ファイルを開いて編集します。
2. 以下の例のように、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。

```
<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>
```

### 4.3. HIBERNATE 接続プロパティの設定

Tomcat プールからの接続を使用するように Hibernate を設定します。Hibernate API を直接使用する場合は、**hibernate.cfg.xml** と似た設定を使用します。

## 手順

1. デプロイメントの `/WEB-INF/classes/META-INF/persistence.xml` ファイルを開いて編集します。
2. 以下の例のように、Hibernate が Tomcat から接続をどのように消費するかを設定します。

```
<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>
```

## 4.4. JDBC データソースの追加

JDBC データソースを使用するように Tomcat を設定します。

## 手順

1. デプロイメントの `/WEB-INF/web.xml` ファイルを開いて編集します。
2. `jdbc/DsWebAppDB` データソースを使用する以下のように、`resource-env-ref` 要素で JDBC データソースを設定します。

```
<?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>
```

## 第5章 RED HAT JBOSS WEB SERVER の HTTP/2 の有効化

ハイパーテキスト転送プロトコルは、インターネットを介して(サーバーやブラウザなどの)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。

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 - 暗黙的な情報を省略して送信されるヘッダーのサイズを縮小します。
- 1つの接続に対する複数の要求および応答。バイナリーフレーミングを使用して応答メッセージを分割します。これは、テキストのフレーミングとは異なります。

Red Hat JBoss Web Server での HTTP/2 の使用:

- TLS (**h2**) での暗号化された接続は **サポートされます**。
- TCP (**h2c**) での暗号化されていない接続は**サポートされません**。

### 要件

- root ユーザーアクセス (Red Hat Enterprise Linux システム)
- Red Hat JBoss Web Server 5.0 以降
- 以下のオペレーティングシステムネイティブライブラリー (利用可能な場合は **jws-5.5.0-application-server-<platform>-<architecture>.zip** によって提供)。  
RHEL-8 ユーザーが JSSE+OpenSSL または APR を実行する必要がある場合、Tomcat-Native が正常に機能するよう使用する必要があります。Tomcat-Native のファイルはネイティブアーカイブディレクトリーにあります。OpenSSL および APR をインストールするには、以下のコマンドを実行します。

```
# yum install openssl
```

```
# yum install apr
```

- SSL が有効になっている HTTP/2 プロトコルをサポートするコネクター。JBoss Web Server 5.5 では、HTTP/2 プロトコルがサポートされるコネクターは次のとおりです。
  - APR ネイティブコネクター (APR)
  - JSSE + OpenSSL (JSSE) のある NIO コネクター
  - JSSE + OpenSSL (JSSE) のある NIO2 コネクター

### 手順

コネクターの HTTP/2 を有効にします。

1. サーバー設定のコネクター **JWS\_HOME/tomcat/conf/server.xml** に、HTTP/2 アップグレードプロトコル (`<UpgradeProtocol className="org.apache.coyote.http2.Http2Protocol" />`) を追加します。  
例を以下に示します。

```
<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** には、HTTP/2 へのアップグレードプロトコルを使用する APR プロトコルのコネクタ定義例が含まれています。

```
<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>
```

2. Red Hat JBoss Web Server を root ユーザーとして再起動し、変更した設定を適用します。
  - a. systemd (Red Hat Enterprise Linux 7) ユーザーの場合:

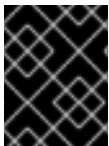
```
# systemctl restart jws5-tomcat.service
```

- b. **startup.sh** を使用して Red Hat JBoss Web Server を実行している Red Hat Enterprise Linux ユーザーの場合:

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

- c. Windows Server ユーザーの場合:

```
# 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。

## 次のステップ

Red Hat JBoss Web Server ログを確認して、または **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します。

- コンソール出力ログ (**JWS\_HOME/tomcat/logs/catalina.out**) を確認して、コネクタが [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確認してください。

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

```
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

- または、**curl** (HTTP2 をサポートする **curl** のバージョン) を使用して検証します。



### 注記

HTTP/2 サポートについて **curl** を確認するには、以下を実行します。

```
$ curl -V
```

```
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
```

- たとえば、HTTP/2 プロトコルが非アクティブである場合:

```
$ curl -I http://<JBoss_Web_Server>:8080/
HTTP/1.1 200
...
```

- ただし、HTTP/2 プロトコルがアクティブである場合、**curl** は以下を返します。

```
$ curl -I https://<JBoss_Web_Server>:8443/
HTTP/2 200
...
```

ここで、<JBoss\_Web\_Server> は変更したコネクターの URI (**example.com** など) で、ポート番号は設定によって異なります。

### その他のリソース

- 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:[IETF: RFC 7540 - Hypertext Transfer Protocol Version 2 \(HTTP/2\)](#)

## 第6章 RED HAT JBOSS WEB SERVER の VAULT

### 6.1. RED HAT JBOSS WEB SERVER 5.5 のパスワード VAULT

**Tomcat-vault** は、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 キーストアに格納するために使用できる Apache Tomcat の **PicketLink vault** エクステンションです。vault を使用することで、キーストアからパスワードやその他の機密文字列を検索することができるため、Tomcat 設定ファイルにクリアテキストパスワードを保存する必要がなくなります。



#### 重要

Vault で CRYPT を使用する方法は、「[CRYPT の使用](#)」を参照してください。

### 6.2. .ZIP アーカイブからの JBOSS WEB SERVER パスワード VAULT のインストール

Tomcat パスワード vault は **jws-5.5.0-application-server.zip** ファイルによって事前にインストールされます。パスワード vault は設定後に使用でき、**JWS\_HOME/tomcat/lib/tomcat-vault.jar** に置かれます。

### 6.3. YUM パッケージマネージャーを使用した RED HAT ENTERPRISE LINUX への JBOSS WEB SERVER パスワード VAULT のインストール

JBoss Web Server が Red Hat Enterprise Linux の RPM からインストールされた場合は、tomcat-vault の JBoss Web Server RPM をインストールする必要があります。

#### 手順

- 以下を実行して、パスワード vault を root ユーザーとしてインストールします。

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

### 6.4. JBOSS WEB SERVER でのパスワード VAULT の有効化

以下の手順では、**JWS\_HOME** を JBoss Web Server インストールへのパスに置き換えます。また、以下のパスではディレクトリー区切り文字として / が使用されます。

#### 手順

1. Tomcat が実行されている場合は停止します。
2. **JWS\_HOME/tomcat/conf/catalina.properties** を編集し、以下の行を追加します。

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

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

パスワード vault を使用するには、最初に Java キーストアを作成する必要があります。



**重要**

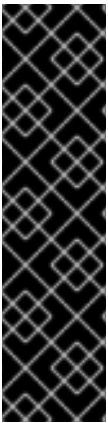
この手順の値は例です。ご使用の環境に固有の値に置き換えてください。

パラメーターの説明は、**keytool -genseckey -help** コマンドを使用します。

**手順**

- **keytool -genseckey** コマンドを使用して Java キーストアを作成します。

```
$ 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
```

**重要**

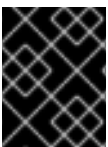
現時点では、キーストアタイプ **PKCS12** は tomcat-vault ではサポートされません。キーストアタイプ **JCEKS** のみがサポートされます。

また、以下のキーストアアルゴリズムには、以下のキーサイズが必要です。

- AES: **-keysize 128**
- DES: **-keysize 56**
- DESede: **-keysize 168**

## 6.6. パスワード VAULT の初期化

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

**重要**

以下の値は例としてのみ使用されます。ご使用の環境に適切な値に置き換えてください。

**手順**

- **tomcat-vault.sh** スクリプトを使用してパスワード vault を初期化します。

```
# JWS_HOME/tomcat/bin/tomcat-vault.sh
```

```
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

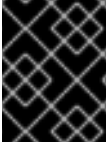
2

```

Tomcat プロパティファイルの出力に注意してください。これは、Tomcat が vault を使用するように設定する必要があるためです。

### 6.6.2. Apache Tomcat の Vault の非対話的な初期化 (サイレント設定)

Apache Tomcat の Vault は、**tomcat-vault.sh** スクリプトに必要な入力を引数として提供して非対話的に作成できます。**vault.properties** ファイルは、**-g, --generate-config** オプションが使用される場合に **tomcat-vault.sh** スクリプトの出力としても作成されます。



## 重要

以下の値は例としてのみ使用されます。ご使用の環境に適切な値に置き換えてください。

### 手順

- **tomcat-vault.sh** スクリプトを使用してパスワード vault を初期化します。

```
$ JWS_HOME/tomcat/bin/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
```

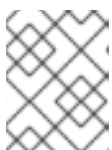
## 6.7. パスワード VAULT を使用するための TOMCAT の設定

### 要件

- Tomcat のパスワード vault が初期化されます。  
Tomcat のパスワード vault の初期化に関する詳細は、「[Initializing password vault for Apache Tomcat interactively](#)」を参照してください。

### 手順

- **JWS\_HOME/tomcat/conf/** で、vault の初期化時に生成された vault 設定を含む **vault.properties** という名前のファイルを作成します。  
以下の値は、「[Initializing password vault for Apache Tomcat interactively](#)」の手順で初期化される vault の例を使用します。



## 注記

**KEYSTORE\_PASSWORD** には vault の初期化時に生成されたマスクされた値を使用する必要があります。

```
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/
```

## 6.8. 外部パスワード VAULT の設定

**tomcat-vault** の **vault.properties** ファイルは、**CATALINA\_BASE/conf/** ディレクトリーの **JWS\_HOME/tomcat/conf/** の外部に保存できます (設定されている場合)。

CATALINA\_BASE ディレクトリーを設定するには、Apache Tomcat Web サイトにある『[Running The Apache Tomcat 9.0 Servlet/JSP Container](#)』の「Advanced Configuration - Multiple Tomcat Instances」セクションの手順に従います。



#### 注記

CATALINA\_BASE のデフォルトの場所は **JWS\_HOME/tomcat/** です (CATALINA\_HOME と呼ばれます)。

#### その他のリソース

CATALINA\_BASE の設定に関する詳細は、以下を参照してください。

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

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

前の手順で使用した vault スクリプトは、機密文字列をパスワード vault に保存するためにも使用されます。スクリプトは対話的または非対話的に実行できます。

文字列をパスワード vault に追加する場合、機密文字列には参照される名前が必要です。パスワード vault では、この名前は **attribute name** と呼ばれ、パスワード自体は **secured attribute** と呼ばれます。

以下の例は、vault スクリプトを非対話的に使用してパスワードを保存する方法を示しています。前の手順で初期化された vault を使用し、機密文字列 **P@SSW0#D** を属性名 **manager\_password** で保存します。

```
$ JWS_HOME/tomcat/bin/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

オプションで、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と、自動的に作成されます。上記の例では、my\_block が使用されます。

## 6.10. TOMCAT 設定で保存された機密文字列の使用

パスワード vault に機密文字列を保存したら、保存された文字列の属性を **#{VAULT::block\_name::attribute\_name::}** として入力して設定ファイルで参照できます。

たとえば、前の手順で保存されたパスワードを使用するには、以下を置き換えます。

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

上記の行を、以下のように置き換えます。

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

その結果、Tomcat 設定ファイルに、パスワードへの参照のみが表示され、実際のパスワードはパスワード vault にのみ保存されます。

## 第7章 SSI フィルターの設定

### 7.1. SSI フィルターの設定

以前のバージョンで行ったように 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

SSI フィルター設定が正しく機能するには、**conf/web.xml** ファイルの以下のブロックをコメント解除します。

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

## 付録A JAVA IPV4/IPV6 プロパティ

### Java プロパティの設定

Java には、IPv4 と IPv6 の設定に使用する 2 つのプロパティがあります。これらは **java.net.preferIPv4Stack** および **java.net.preferIPv6Addresses** です。

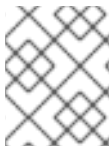
#### java.net.preferIPv4Stack (default: false)

IPv6 が利用できる場合は、デフォルトで基礎となるネイティブソケットが IPv6 ソケットになります。このソケットにより、アプリケーションが IPv4 および IPv6 ホストからの接続を許可できるようになります。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、アプリケーションが IPv6 のみのホストと通信することはできません。

#### java.net.preferIPv6Addresses (デフォルト: 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。192.168.1.1 など、IPv4 アドレス表現に依存するアプリケーションの場合、可能な場合は IPv4 で IPv6 アドレスを使用するように、このプロパティを **true** に設定します。

これらのプロパティを Tomcat に渡すには、**JWS\_HOME/tomcat/bin/setenv.\*** ファイルに **CATALINA\_OPTS** を設定します。



#### 注記

**JWS\_HOME/tomcat/bin/setenv.sh** または **JWS\_HOME/tomcat/bin/setenv.bat** ファイルが存在しない場合は、作成する必要があります。

Linux の場合:

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

Windows の場合:

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

### Tomcat バインディングの設定

Tomcat バインディングは、IPv6 アドレスを使用して **JWS\_HOME/tomcat/conf/server.xml** で設定できます。

- Tomcat バインディングアドレスを指定します。  
<Server ... address="TOMCAT\_BINDING\_ADDRESS">
- HTTP コネクターアドレスを指定します。  
<Connector protocol="HTTP/1.1" ... address="HTTP\_CONNECTOR\_ADDRESS">
- AJP コネクターアドレスを指定します。  
<Connector protocol="AJP/1.3" ... address="AJP\_CONNECTOR\_ADDRESS">