# Red Hat JBoss Core Services 2.4.57

## Apache HTTP Server インストールガイド

Red Hat JBoss ミドルウェア製品との使用

Last Updated: 2024-02-08

# Red Hat JBoss Core Services 2.4.57 Apache HTTP Server インストールガイド

Red Hat JBoss ミドルウェア製品との使用

## 法律上の通知



## 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 Core Services Apache HTTP Server をインストール、アップグレード、および設定します。

# 目次

# RED HAT JBOSS CORE SERVICES 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 プロパティー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 JBCS APACHE HTTP SERVER のインストールの概要

Red Hat JBoss Core Services (JBCS) は、Apache HTTP Server をはじめとする、各種 Red Hat JBoss ミドルウェア製品で使用できる補助ソフトウェアのコレクションを提供します。Red Hat は、迅速な更新の配布と一貫性のある更新作業を実現するために、この補助ソフトウェアを JBCS 配下にパッケージ化しています。

JBCS でサポートされているコンポーネントの完全なリストは、Core Services Apache HTTP Server コンポーネントの詳細 を記載している Web ページで確認してください。

**注記**

Core Services Apache HTTP Server コンポーネントの詳細 の Web ページにアクセスする前に、有効な Red Hat サブスクリプションがあり、Red Hat カスタマーポータルにログインしていることを確認してください。

## 1.1. JBCS APACHE HTTP SERVER

Red Hat JBoss Core Services (JBCS) は、複数の Red Hat JBoss ミドルウェア製品が使用する Apache HTTP Server のディストリビューションを提供します。Apache HTTP Server は、Web クライアントが Hypertext Transfer Protocol (HTTP) 経由で送信するリクエストを処理します。

**JBoss ミドルウェア製品用の Apache HTTP Server ディストリビューション**
古い JBoss 製品リリースでは、各 JBoss ミドルウェア製品が Apache HTTP サーバーの個別のディストリビューションを提供していました。以下の製品バージョン以降、JBoss ミドルウェアの各製品は、Apache HTTP Server の JBCS ディストリビューションを使用します。

- Red Hat JBoss Enterprise Application Platform (JBoss EAP) 7.0 以降
- Red Hat JBoss Web Server 3.1 以降

**Apache HTTP Server の JBCS ディストリビューションと RHEL ディストリビューションの違い**
JBCS と Red Hat Enterprise Linux (RHEL) では、Apache HTTP Server を別々に配布しています。

**重要**

RHEL 9 では、JBCS は Apache HTTP Server の RPM ディストリビューションを提供しません。JBCS は、RHEL 9 システム用の Apache HTTP Server アーカイブファイルディストリビューションのみを提供します。

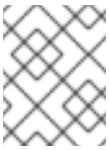
以前の RHEL バージョンの JBCS リリースとは異なり、RHEL 9 システム用 Apache HTTP Server の JBCS ディストリビューションは、Apache HTTP Server **httpd** パッケージの RHEL ディストリビューションをベースにしています。JBCS は、Apache HTTP Server の複数のインスタンスを同時に実行する機能をサポートするために、RHEL 9 上でアーカイブファイルディストリビューションを提供します。

以下に記載した、JBCS と RHEL が提供する Apache HTTP Server ディストリビューションの違いを考慮してください。

**RHEL バージョン 7、8**

- アーカイブファイルまたは RPM パッケージから JBCS Apache HTTP Server をインストールできます。RHEL Apache HTTP Server は、RPM パッケージからのみインストールできます。

- JBCS Apache HTTP Server のみが、負荷分散 HTTP コネクター **mod_jk** および **mod_proxy_cluster** を提供します。RHEL Apache HTTP Server は、これらのモジュールを提供しません。



**注記**

JBCS 2.4.37 以前のリリースでは、**mod_proxy_cluster** コネクターの名前は **mod_cluster** でした。

- RHEL 7 では、JBCS Apache HTTP Server のみが **mod_proxy_uwsgi** モジュールを提供します。RHEL 8 以降では、Apache HTTP Server の JBCS ディストリビューションと RHEL ディストリビューションの両方が **mod_proxy_uwsgi** モジュールを提供します。

**RHEL 9 の場合**

- RHEL 7 および RHEL 8 の JBCS リリースとは異なり、RHEL 9 の JBCS リリースは、Apache HTTP Server **httpd** パッケージの RHEL ディストリビューションをベースにしています。したがって、RHEL 9 上の JBCS は、それより前の RHEL バージョンにおける Apache HTTP Server の JBCS ディストリビューションと比較して動作が異なります。詳細は、異なる RHEL バージョンの JBCS ディストリビューションにおける動作の違い を参照してください。

- JBCS は、Apache HTTP Server のアーカイブファイルディストリビューションのみを提供します。Apache HTTP Server を RPM パッケージからインストールする場合、選択できる方法は 1 つだけで、Application Streams を使用して **httpd** パッケージの RHEL ディストリビューションをインストールします。

- JBCS が提供する Apache HTTP Server のバージョンは、RHEL が Application Streams 機能を通じて提供する Apache HTTP Server のバージョンとは異なります。

- Apache HTTP Server の JBCS および RHEL ディストリビューションは、**mod_jk** コネクターと **mod_proxy_cluster** コネクターの同一コピーを提供します。

**すべての RHEL バージョンの場合**

- JBCS Apache HTTP Server は、最上位の **jbcs-httpd24-2.4/httpd** インストールディレクトリーを使用します。RHEL Apache HTTP Server は、**httpd** パッケージのインストールに **/etc/httpd**、**usr/share/httpd**、**var/log/httpd** などの標準 RHEL ディレクトリーを使用します。

- **groupinstall** オプションを使用してアーカイブファイルまたは RPM パッケージから Apache HTTP Server の JBCS ディストリビューションをインストールすると、**mod_jk** コネクターと **mod_proxy_cluster** コネクターも自動的にインストールされます。

- JBCS Apache HTTP Server は、**mod_php** モジュールを提供またはサポートしていません。RHEL Apache HTTP Server のみが、**mod_php** モジュールをサポートしています。

### 異なる RHEL バージョン上の JBCS ディストリビューションにおける動作の違い

RHEL 7 または RHEL 8 上の JBCS 2.4.57 とは異なり、RHEL 9 システム用の JBCS 2.4.57 ディストリビューションは、Apache HTTP Server **httpd** パッケージの RHEL ディストリビューションをベースにしています。RHEL 9 以降で Red Hat による **httpd** パッケージの配布方法が変更されたことで、さらぶに一貫性のある合理的なユーザーエクスペリエンスを Apache HTTP Server ユーザーに提供できるようになりました。

この違いのため、RHEL 9 の JBCS 2.4.57 は、それより前の RHEL バージョンの JBCS 2.4.57 と比較して動作が異なります。

次のガイドラインを考慮してください。

- RHEL 9 は、**mod_security** モジュールは、ガベージコレクションの頻度を指定する **SecCollectionGCFrequency** ディレクティブをサポートしません。RHEL 7 および RHEL 8 で JBCS が 提供する **mod_security** モジュールは、**SecCollectionGCFrequency** ディレクティブをサポートします。

- RHEL 9 は、**mod_deflate** モジュールは、応答が圧縮されるときに ETag ヘッダーを変更する方法を指定する **DeflateAlterEtag** ディレクティブをサポートしません。JBCS が RHEL 7 および RHEL 8 で提供する **mod_deflate** モジュールは、**DeflateAlterEtag** ディレクティブをサポートします。

- RHEL 9 では、**httpd.conf.sample** ファイルに次の内容は含まれません。

  - サーバーがデーモンのプロセス ID を記録するファイルを指定するためのデフォルトの **PidFile** ディレクティブ

  - 特定のファイル名エクステンションを特定のコンテンツ言語にマッピングするための **mod_mime** セクション内の **AddLanguage** ディレクティブリスト

  - Web ベースの分散オーサリングおよびバージョン管理 (WebDav) に使用する **web_dav** モジュールの設定セクション

  JBCS が RHEL 7 および RHEL 8 で提供する **httpd.conf.sample** ファイルには、前述の内容がすべて含まれています。

## 1.2. JBCS APACHE HTTP SERVER でサポートされているオペレーティングシステムとインストール方法

Red Hat JBoss Core Services (JBCS) は、Red Hat Enterprise Linux (RHEL) および Windows Server オペレーティングシステムのさまざまなバージョンに対応する Apache HTTP Server のディストリビューションを提供します。

サポートされているオペレーティングシステムに JBCS Apache HTTP Server をインストールする場合は、次のガイドラインを考慮してください。

- サポートされているすべての RHEL および Windows Server バージョンに、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、JBCS Apache HTTP Server をインストールできます。

- RHEL バージョン 7 および 8 では、Red Hat Package Manager (RPM) パッケージを使用して JBCS Apache HTTP Server をインストールできます。

- RHEL 9 で、RPM パッケージから Apache HTTP Server をインストールする場合は、Application Streams を使用して Apache HTTP Server の RHEL ディストリビューションをインストールする必要があります。RHEL 9 では、RPM パッケージを使用して JBCS Apache HTTP Server を RHEL 9 にインストールすることは **できません**。

**関連情報**

- Core Services HTTP Server でサポートされる設定 の Web ページ

## 1.3. 既存の JBCS インストールを 2.4.57 リリースにアップグレードする方法

Red Hat JBoss Core Services (JBCS) 2.4.51 以前をインストールしている場合は、既存の JBCS インストールを最新の 2.4.57 リリースにアップグレードできます。JBCS をアップグレードする手順は、製品をアーカイブファイルと RPM パッケージのどちらからインストールしたかにより異なります。

### 1.3.1. アーカイブファイルからインストールした既存 JBCS インストールのアップグレード

JBCS Apache HTTP Server 2.4.51 以前をアーカイブファイルからインストールしている場合は、最新の 2.4.57 リリースにアップグレードできます。

アップグレードプロセスには、以下の手順が含まれます。

1. Apache HTTP Server 2.4.57 のインストール
2. Apache HTTP Server 2.4.57 の設定
3. 以前の Apache HTTP Server バージョンの削除

**前提条件**

- Red Hat Enterprise Linux (RHEL) を使用している場合は、root ユーザーアクセス権がある。
- Windows Server を使用している場合は、管理アクセス権がある。
- アーカイブファイルからインストールした JBCS Apache HTTP Server 2.4.51 以前の既存インストールがある。

**手順**

1. Apache HTTP Server 2.4.51 の実行中のインスタンスをすべてシャットダウンします。
2. Apache HTTP Server 2.4.51 のインストールファイルと設定ファイルをバックアップします。
3. 現在のシステムのアーカイブファイルインストール方法を使用して、Apache HTTP Server 2.4.57 をインストールします。詳細は、このセクションの最後にある 関連情報 を参照してください。
4. 設定を Apache HTTP Server バージョン 2.4.51 からバージョン 2.4.57 に移行します。

**注記**

JBCS 設定ファイルは、Apache HTTP Server 2.4.51 リリース以降に変更されている可能性があります。2.4.57 バージョンの設定ファイルは、別のバージョン (Apache HTTP Server 2.4.51 など) の設定ファイルで上書きするのではなく、更新してください。

5. Apache HTTP Server 2.4.51 ルートディレクトリーを削除します。

**関連情報**

- アーカイブファイルから RHEL に JBCS Apache HTTP サーバーをインストールする

- Windows Server に JBCS Apache HTTP サーバーをインストールする

### 1.3.2. RPM パッケージからインストールされた既存 JBCS インストールのアップグレード

RPM パッケージから JBCS Apache HTTP Server 2.4.51 以前をインストールしている場合は、**yum groupupdate** コマンドを使用して最新の 2.4.57 リリースにアップグレードできます。

**前提条件**

- RHEL 7 または RHEL 8 上で RPM パッケージからインストールした JBCS Apache HTTP Server 2.4.51 以前の既存インストールがある。

**手順**

- root ユーザーとして以下のコマンドを実行します。

```
# yum groupupdate jbcs-httpd24
```

**関連情報**

- RPM パッケージから RHEL 7 または RHEL 8 に JBCS Apache HTTP サーバーをインストールする

## 1.4. RHEL 7 と RHEL 8 の主な相違点

このセクションでは、Red Hat Enterprise Linux (RHEL) 8 で導入された主な変更点の概要を説明します。

**削除されたセキュリティー機能**

RHEL 7 では数字だけのユーザー名およびグループ名がすべて非推奨となり、RHEL 8 ではサポートが完全に削除されました。

**メモリー管理**

RHEL 7 での既存のメモリーバスには、48/46 ビットの仮想または物理のメモリーアドレス容量があり、Linux カーネルが、4 つのレベルのページテーブルを実装して、物理アドレスへの仮想アドレスを管理します。アドレス範囲の拡張に伴い、RHEL 8 のメモリー管理は 5 レベルのページテーブルの実装をサポートし、拡張されたアドレス範囲に対応できるようにしました。RHEL 8 では、5 レベルページテーブルのサポートは、システムがこの機能をサポートしている場合でも、デフォルトで無効になっています。

**XFS に対応**

RHEL 7 は、読み取り専用モードでのみ、共有コピーオンライトのデータエクステントを持つ XFS ファイルシステムをマウントできます。RHEL 8 の場合、XFS ファイルシステムは、共有コピーオンライトのデータエクステント機能に対応します。この機能により、2 つ以上のファイルが共通のデータブロックセットを共有できます。

**NFS の設定**

RHEL 7 の場合、NFS 設定は **/etc/sysconfig/nfs** ファイルにあります。RHEL 8 の場合、NFS 設定は **/etc/nfs.conf** ファイルにあります。

**関連情報**

- RHEL 8 の導入における検討事項

## 1.5. RHEL 8 と RHEL 9 の主な相違点

このセクションでは、Red Hat Enterprise Linux (RHEL) 9 で導入された主な変更点の概要を説明します。

### Application Streams の機能拡張

RHEL 8 では、**Application Streams** と呼ばれる機能が導入されました。RHEL は Application Streams を使用して、アプリケーション、ランタイム言語、データベースなどのユーザー空間コンポーネントの複数バージョンを、コアオペレーティングシステムパッケージよりも頻繁に配信および更新します。各 Application Streams はコンポーネントの特定バージョンを表し、Application Streams 内の各コンポーネントには定義されたライフサイクルがあります。ユーザーは Application Streams を使用することで、プラットフォームやデプロイメントの基礎となる安定性に影響を与えることなく、特定のユースケースやワークロードの要件に適したコンポーネントバージョンを使用できる柔軟性を得ることができます。

Red Hat は RHEL 8 で、RPM パッケージ、モジュール (パッケージグループ)、ソフトウェアコレクションの組み合わせとして、Application Streams のコンテンツをパッケージ化しました。RHEL 9 は、標準の **dnf install** コマンドを使用して RPM パッケージとしてインストールできる初期の Application Stream バージョンを提供することにより、Application Streams 機能をさらに強化しています。

### Apache コネクターとロードバランサーの可用性

RHEL 9 は、Web クライアント要求をバックエンドアプリケーションサーバーに負荷分散するための Apache Tomcat コネクター (**mod_jk**) および JBoss HTTP コネクター (**mod_proxy_cluster**) のディストリビューションを提供します。**mod_jk** および **mod_proxy_cluster** の RHEL ディストリビューションは、これらのモジュールの JBCS ディストリビューションと同じです。

Apache HTTP Server の RHEL 9 ディストリビューションをインストールしても、**mod_jk** モジュールと **mod_proxy_cluster** モジュールは自動的にインストールされません。RHEL 9 上の RPM パッケージから **mod_jk** および **mod_proxy_cluster** をインストールする方法についての詳細は Apache HTTP Server コネクターおよび負荷分散ガイド を参照してください。

### 関連情報

- RHEL 9 の採用における考慮事項

## 1.6. 関連情報 (または次の手順)

- アーカイブファイルから RHEL に JBCS Apache HTTP サーバーをインストールする
- RPM パッケージから RHEL 7 または RHEL 8 に JBCS Apache HTTP サーバーをインストールする
- Windows Server に JBCS Apache HTTP サーバーをインストールする
- Application Streams を使用した RHEL 9 への Apache HTTP サーバーのインストール

# 第2章 アーカイブファイルから RHEL に JBCS APACHE HTTP サーバーをインストールする

Red Hat Enterprise Linux (RHEL) バージョン 7、8、9 では、アーカイブファイルからインストールできる Apache HTTP Server のディストリビューションが Red Hat JBoss Core Services (JBCS) により提供されます。Red Hat カスタマーポータルの ソフトウェアダウンロード ページからアーカイブファイルをダウンロードして展開できます。元の 2.4.57 リリースのベースアーカイブファイルをインストールする必要があります。最新のサービスパックリリースがある場合は、それをインストールすることもできます。

アーカイブファイルから Apache HTTP Server を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 Apache HTTP Server を管理したりすることができます。

**注記**

2.4.57 Service Pack 2 リリース以降では、JBDC は RHEL 9 上のアーカイブファイルからの Apache HTTP Server 2.4.57 のインストールをサポートしています。RHEL 9 上の JBCS Apache HTTP Server 2.4.57 インストールのベースアーカイブファイルは **Red Hat JBoss Core Services Apache HTTP Server 2.4.57 Patch 02 for RHEL 9 x86_64** です。

## 2.1. RHEL での APACHE HTTP SERVER アーカイブファイルのダウンロードとデプロイメント

Red Hat カスタマーポータルの ソフトウェアダウンロード ページから、Apache HTTP Server アーカイブファイルをダウンロードできます。使用している Red Hat Enterprise Linux (RHEL) のバージョンによって、アーカイブファイルをダウンロードする手順が若干異なります。

**注記**

目的のインストールディレクトリーへの書き込みアクセス権があれば、root 以外の権限でアーカイブファイルをインストールできます。

**前提条件**

- **elinks**、**krb5-workstation**、および **mailcap** パッケージがインストールされている。
  これらのパッケージをインストールする場合は、root ユーザーとして次のコマンドを入力します。

```
# yum install elinks krb5-workstation mailcap
```

**手順**

1. ブラウザーを開き、Red Hat カスタマーポータルの Software Downloads ページにログインします。
2. **Product** ドロップダウンメニューから、**Apache HTTP Server** を選択します。
3. **Version** ドロップダウンメニューから、正しい JBCS バージョンを選択します。
4. 使用している RHEL バージョンに応じて、以下のいずれかの手順を実行します。

- RHEL 7 を使用している場合は、**Release** タブで、**Red Hat JBoss Core Services Apache HTTP Server 2.4.57 for RHEL 7 x86_64** ファイルの横にある **Download** をクリックします。
- RHEL 8 を使用している場合は、**Release** タブで、**Red Hat JBoss Core Services Apache HTTP Server 2.4.57 for RHEL 8 x86_64** ファイルの横にある **Download** をクリックします。
- RHEL 9 を使用している場合は、**Security Advisories** タブをクリックします。続いて、**Red Hat JBoss Core Services Apache HTTP Server 2.4.57 Patch 02 for RHEL 9 x86_64** ファイルの横にある **Download** をクリックします。

**注記**

**Red Hat JBoss Core Services Apache HTTP Server 2.4.57 Patch 02 for RHEL 9 x86_64** ファイルは、RHEL 9 に JBCS Apache HTTP Server 2.4.57 をインストールするためのベースアーカイブファイルです。

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。

**注記**

RHEL システムの場合は、Apache HTTP サーバーを /**opt**/ ディレクトリーにインストールします。

アーカイブファイルを展開すると、Apache HTTP サーバーの最上位の **jbcs-httpd24-2.4/httpd** ディレクトリーが自動的に作成されます。本書では、**jbcs-httpd24-2.4/httpd** ディレクトリーを **HTTPD_HOME** と呼びます。

6. 最新のサービスパックリリースがある場合は、次の手順を実行してインストールします。
   a. Software Downloads ページで、**Security Advisories** タブをクリックします。
   b. **Security Advisories** タブで、お使いの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最新の JBCS Apache HTTP Server 2.4.57 パッチアーカイブファイルの横にある **Download** をクリックします。
   たとえば、RHEL 8 に Apache HTTP Server 2.4.57 の Service Pack **X** リリースをインストールする場合は、**Red Hat JBoss Core Services Apache HTTP Server 2.4.57 Patch X for RHEL 8 x86_64** ファイルの横にある **Download** をクリックします。

**注記**

サービスパックのリリースは累積的です。最新のサービスパックリリースをダウンロードすると、以前のサービスパックリリースも自動的にインストールされます。

## 2.2. 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 APACHE HTTP SERVER 設定

RHEL 上のアーカイブファイルから JBCS Apache HTTP サーバーをインストールすると、コマンドラインから直接 Apache HTTP サーバーを起動および停止できます。コマンドラインから Apache HTTP Server を実行する前に、以下の一連の設定作業を行う必要があります。

- Apache ユーザーを作成する
- SSL サポートを無効または有効にする
- Apache HTTP Server のインストール後のスクリプトを実行する

### 2.2.1. Apache ユーザーの作成

コマンドラインから Apache HTTP Server を初めて実行する前に、**apache** ユーザーアカウントとグループを作成する必要があります。また、ユーザーが Apache HTTP Server を実行できるように、Apache ディレクトリーの所有権を **apache** ユーザーに割り当てる必要があります。

**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**

- アーカイブファイルから Apache HTTP Server をインストールしている。

**手順**

1. コマンドラインで、**HTTPD_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 **apache** ユーザーグループを作成します。

```
# groupadd -g 48 -r apache
```

3. 以下のコマンドを実行して、**apache** ユーザーグループに **apache** ユーザーを作成します。

```
# /usr/sbin/useradd -c "Apache" -u 48 -g apache -s /sbin/nologin -r apache
```

4. 以下のコマンドを実行して、Apache ディレクトリーの所有権を **apache** ユーザーに割り当てます。

```
# chown -R apache:apache *
```

**検証**

- 以下のコマンドを実行して、**apache** 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
```

### 2.2.2. SSL サポートの無効化または有効化

Apache HTTP Server を実行する前に、SSL 設定ファイルの名前を変更して、SSL サポートを無効または有効にすることを選択できます。Apache HTTP Server は、デフォルトで SSL をサポートします。

**手順**

1. **HTTPD_HOME/conf.d/** ディレクトリーに移動します。

2. SSL を有効または無効にするには、次のいずれかの手順を実行します。
   - SSL を無効にするには、**ssl.conf** の名前を **ssl.conf.disabled** に変更します。
   - SSL を再度有効にするには、**ssl.conf.disabled** の名前を **ssl.conf** に変更します。

### 2.2.3. Apache HTTP Server のインストール後のスクリプトを実行する

コマンドラインから Apache HTTP Server を初めて実行する前に、Apache HTTP Server のインストール後のスクリプトを実行する必要があります。

**手順**

1. コマンドラインで、**HTTPD_HOME** ディレクトリーに移動します。
2. 以下のコマンドを入力します。

```
./.postinstall
```

## 2.3. 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 APACHE HTTP SERVER の起動

RHEL 上のアーカイブファイルから JBCS Apache HTTP サーバーをインストールすると、コマンドラインから直接 Apache HTTP サーバーを起動できます。

**前提条件**

- **Apache** ユーザーを作成 している。
- SSL サポートを無効化または再度有効化 している。
- Apache HTTP Server のインストール後のスクリプトを実行 している。

**手順**

1. コマンドラインで、**HTTPD_HOME/sbin/** ディレクトリーに移動します。
2. root ユーザーとして以下のコマンドを実行します。

```
./apachectl start
```

## 2.4. 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 APACHE HTTP SERVER の停止

RHEL 上のアーカイブファイルから JBCS Apache HTTP サーバーをインストールすると、コマンドラインから実行中の Apache HTTP サーバーのインスタンスを直接停止できます。

**前提条件**

- Apache HTTP Server を開始 している。

**手順**

1. コマンドラインで、**HTTPD_HOME/sbin/** ディレクトリーに移動します。
2. root ユーザーとして以下のコマンドを実行します。

```
./apachectl stop
```

## 2.5. ROOT 権限なしでコマンドラインから APACHE HTTP SERVER を実行

RHEL 上のアーカイブファイルから JBCS Apache HTTP サーバーをインストールする場合、root 権限のないユーザーとしてコマンドラインから Apache HTTP サーバーを起動できます。この場合、**apache** ユーザーなどの非 root ユーザーアカウントを使用できます。

**手順**

1. Apache HTTP Server のすべてのインスタンスを停止します。

```
pkill httpd
```

2. **HTTPD_HOME/conf/httpd.conf** ファイルで、**http** リッスンポートを 1024 より大きい値に設定します。

```
Listen 2080
ServerName <hostname>:2080
```

3. **HTTPD_HOME/conf.d/ssl.conf** ファイルで、**https** リッスンポートを 1024 より大きい値に設定します。

```
Listen 2443
```

4. **logs** ディレクトリーの所有権を変更します。

```
chown -R apache:apache HTTPD_HOME/logs/
```

5. **run** ディレクトリーの所有権を変更します。

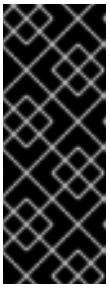
```
chown -R apache:apache HTTPD_HOME/var/run/
```

6. **httpd** が、**root** ユーザーと **apache** ユーザーではなく、**apache** ユーザーでのみ実行されていることを確認します。

```
$ ps -eo euser,egroup,comm | grep httpd
```

このコマンドは、次のタイプの出力を生成します。

```
apache   apache   httpd
apache   apache   httpd
apache   apache   httpd
...
```



**重要**

**apache** ユーザーのファイルパーミッションを制限し、SELinux を有効化します。これにより、以下のようなシナリオを防ぐことができます。

- Web サイト利用者によるファイルやディレクトリーへの不正アクセスや改変
- Apache HTTP Server 設定ファイルへの不要な変更

## 2.6. 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 APACHE HTTP SERVER を管理する

RHEL 上のアーカイブファイルから JBCS Apache HTTP サーバーをインストールすると、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。Apache HTTP Server をシステムデーモンと併用すると、システム起動時に Apache HTTP Server サービスを開始する方法を使用できるようになります。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。

RHEL バージョン 7、8、9 の場合、デフォルトのシステムデーモンは **systemd** です。

### 前提条件

- アーカイブファイルから Apache HTTP Server をインストールしている。

### 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 Apache HTTP 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 HTTPD_HOME
# sh httpd/.postinstall.systemd
```

3. **systemd** を使用して Apache HTTP サーバーを制御するには、root ユーザーとして次のコマンドのいずれかを入力します。
   - システム起動時に Apache HTTP Server サービスを開始するには、以下を入力します。

```
# systemctl enable jbcs-httpd24-httpd.service
```

   - Apache HTTP Server を起動するには、以下を入力します。

```
# systemctl start jbcs-httpd24-httpd.service
```

   - Apache HTTP Server を停止するには、以下を入力します。

```
# systemctl stop jbcs-httpd24-httpd.service
```

- Apache HTTP Server のステータスを確認するには、以下を入力します。

```
# systemctl status jbcs-httpd24-httpd.service
```

**注記**

すべてのユーザーは **systemctl status** コマンドを実行できます。

**重要**

**.postinstall.systemd** スクリプトの影響を受ける変更を元に戻すには、次のコマンドを入力します。

```
# cd HTTPD_HOME
# sh httpd/.postinstall.services.cleanup
```

**systemd** の使用について、詳しくは **関連情報** リンクを参照してください。

**関連情報**

- RHEL 7: System Administrator's Guide: Managing System Services
- RHEL 8: 基本的なシステム設定: systemctl を使用したシステムサービスの管理
- RHEL 9: 基本的なシステム設定: systemctl を使用したシステムサービスの管理

## 2.7. APACHE HTTP SERVER の SELINUX ポリシー

Security-Enhanced Linux (SELinux) ポリシーを使用して、Apache HTTP 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. 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。

Apache HTTP Server インストールの **jbcs-httpd24-httpd-selinux** パッケージは、**mod_proxy_cluster** 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 |
|---|---|---|
| **mod_proxy_cluster** | 2 つのポート (**TCP** の **6666**、**UDP** の **23364**) は、**httpd** プロセスが使用できるように **httpd_port_t** に追加されます。 | インストール後のスクリプトは、**/var/cache/mod_proxy_cluster** のコンテキストマッピングを設定し、**httpd** プロセスがこの場所に書き込みできるようにします。 |

**関連情報**

- RHEL 7: SELinux ユーザーおよび管理者ガイド
- RHEL 8: SELinux の使用
- RHEL 9: SELinux の使用

### 2.7.2. Apache HTTP 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。ルートの Apache HTTP Server フォルダーには、**.postinstall.selinux** ファイルが格納されています。必要に応じて、**.postinstall.selinux** スクリプトを実行できます。

**重要**

デフォルトでは、Apache HTTP Server が提供する SELinux ポリシーはアクティブではなく、Apache HTTP Server プロセスは **unconfined_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 ポリシーを有効化しないと選択した場合は、**apache** ユーザーのファイルアクセスを制限して、**apache** ユーザーが Apache HTTP Server ランタイムに必要なファイルとディレクトリーにのみアクセスできるようにします。

**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 <httpd_home>
sh .postinstall.selinux
```

3. SELinux モジュールを作成してインストールします。

```
cd <httpd_home>/selinux/
make -f /usr/share/selinux/devel/Makefile
semodule -i jbcs-httpd24-httpd.pp
```

4. Apache HTTP Server の SELinux コンテキストを適用します。

```
restorecon -r <httpd_home>
```

5. Apache HTTP Server に必要なポートへの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 6666
semanage port -a -t http_port_t -p udp 23364
```

6. Apache HTTP Server サービスを起動します。

```
<httpd_home>/sbin/apachectl start
```

7. **httpd_t** 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 -eZ | grep httpd | head -n1

unconfined_u:unconfined_r:httpd_t:s0-s0:c0.c1023 2864 ? 00:00:00 httpd
```

8. httpd 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 <httpd_home>/logs/
```

# 第3章 RPM パッケージから RHEL 7 または RHEL 8 に JBCS APACHE HTTP サーバーをインストールする

Red Hat Enterprise Linux (RHEL) バージョン 7 および 8 では、RPM パッケージからインストールできる Apache HTTP Server のディストリビューションが Red Hat JBoss Core Services (JBCS) により提供されます。JBCS Apache HTTP 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。RPM パッケージから Apache HTTP Server をインストールすると、Apache HTTP Server がサービスとしてインストールされます。

**重要**

JBCS は RHEL バージョン 7 および 8 向けにのみ、Apache HTTP Serverの RPM ディストリビューションを提供します。JBCS は、RHEL 9 向けには Apache HTTP Server の RPM ディストリビューションを提供しません。

RHEL 9 上の RPM パッケージから Apache HTTP Server をインストールする場合は、RHEL の Application Streams 機能を使用する必要があります。詳細は、Application Streams を使用した RHEL 9 への Apache HTTP サーバーのインストール を参照してください。

## 3.1. RHEL へのサブスクリプションのアタッチ

Apache HTTP 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 Enterprise Linux (RHEL) にサブスクリプションをアタッチする必要があります。サブスクリプションをアタッチするには、システムを Red Hat Subscription Management に登録し、それぞれのコンテンツ配信ネットワーク (CDN) リポジトリーにサブスクライブします。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 リポジトリーを提供していることを確認することができます。

**手順**

1. システムを Red Hat Subscription Management に登録するには:
   a. Red Hat Subscription Management Web ページにログインします。
   b. **System** タブをクリックします。
   c. 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。
   d. **Details** タブから **Subscriptions** タブに移動してから、**Attach Subscriptions** をクリックします。
   e. 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。
2. お使いのオペレーティングシステムのバージョンに対応した Apache HTTP Server CDN リポジトリーにサブスクライブするには、root ユーザーで次のコマンドを入力します。

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

**注記**

RHEL 7 を使用している場合は、**<repository>** を **jb-coreservices-1-for-rhel-7-server-rpms** に置き換えます。

RHEL 8 を使用している場合は、**<repository>** を **jb-coreservices-1-for-rhel-8-x86_64-rpms** に置き換えます。

**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** の下では、 **Red Hat JBoss Core Services** が必要です。

インストールされている RHEL バージョンの登録について、詳しくは **関連情報** リンクを参照してください。

**関連情報**

- RHEL 7: インストールガイド: サブスクリプションマネージャー
- RHEL 8: 基本的なシステム設定: システムの登録とサブスクリプションの管理。

## 3.2. YUM を使用した RPM パッケージからの APACHE HTTP 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、RHEL 7 または RHEL 8 の RPM パッケージから JBCS Apache HTTP Server をインストールできます。

**前提条件**

- RHEL にサブスクリプションをアタッチ している。

**手順**

- root ユーザーで以下のコマンドを実行し、Apache HTTP Server をインストールします。

```
# yum groupinstall jbcs-httpd24
```

## 3.3. RPM からインストールした場合の APACHE HTTP SERVER インストールの設定

RPM パッケージから Apache HTTP Server をインストールすると、Apache HTTP Server を実行する前に SSL サポートをオプションで削除することができます。Apache HTTP Server は、デフォルトで SSL をサポートします。**mod_ssl** パッケージを削除することで、SSL サポートを削除することを選択できます。

**手順**

- コマンドラインで、root ユーザーとして次のコマンドを入力します。

```
# yum remove jbcs-httpd24-mod_ssl
```

## 3.4. RPM からインストールした場合のコマンドラインからの APACHE HTTP SERVER の起動

JBCS Apache HTTP Server を RPM パッケージからインストールする場合は、コマンドラインを使用して Apache HTTP Server を起動できます。

**手順**

- コマンドラインで、root ユーザーとして Apache HTTP Server サービスを起動します。

```
# systemctl start jbcs-httpd24-httpd.service
```

## 3.5. RPM からインストールした場合にコマンドラインから APACHE HTTP SERVER を停止する

JBCS Apache HTTP Server を RPM パッケージからインストールする場合は、コマンドラインを使用して Apache HTTP Server を停止できます。

**手順**

- コマンドラインで、root ユーザーとして Apache HTTP Server サービスを停止します。

```
# systemctl stop jbcs-httpd24-httpd.service
```

## 3.6. システム起動時に APACHE HTTP SERVER サービスが起動するように設定する

RPM パッケージから JBoss Core Services Apache HTTP Server をインストールする場合は、Apache HTTP Server サービスがシステム起動時に起動するように設定できます。

**手順**

- システムの起動時に Apache HTTP Server サービスを開始できるようにするには、root ユーザーとして次のコマンドを入力します。

```
# systemctl enable jbcs-httpd24-httpd.service
```

## 3.7. APACHE HTTP SERVER の SELINUX ポリシー

Security-Enhanced Linux (SELinux) ポリシーを使用して、Apache HTTP 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. 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。

Apache HTTP Server インストールの **jbcs-httpd24-httpd-selinux** パッケージは、**mod_proxy_cluster** 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 |
|---|---|---|
| **mod_proxy_cluster** | 2 つのポート (**TCP** の **6666**、**UDP** の **23364**) は、**httpd** プロセスが使用できるように **httpd_port_t** に追加されます。 | インストール後のスクリプトは、**/var/cache/mod_proxy_cluster** のコンテキストマッピングを設定し、**httpd** プロセスがこの場所に書き込みできるようにします。 |

**関連情報**

- RHEL 7: SELinux ユーザーおよび管理者ガイド
- RHEL 8: SELinux の使用

### 3.7.2. Apache HTTP Server RPM インストールの SELinux ポリシーを有効化する

RPM パッケージから JBoss Core Services Apache HTTP Server をインストールする場合は、**jbcs-httpd2.4-httpd-selinux** パッケージにより Apache HTTP Server 用の SELinux ポリシーが提供されます。**jbcs-httpd2.4-httpd-selinux** パッケージは、**jb-coreservices-1-for-rhel-7-server-rpms** および **jb-coreservices-1-for-rhel-8-x86_64-rpms** コンテンツ配信ネットワーク (CDN) リポジトリーから利用可能です。

**手順**

- 使用している RHEL バージョンの **jbcs-httpd2.4-httpd-selinux** パッケージをインストールします。

# 第4章 WINDOWS SERVER に JBCS APACHE HTTP サーバーをインストールする

Red Hat カスタマーポータルの ソフトウェアダウンロード ページにあるダウンロード可能なアーカイブファイルから Windows Server に JBCS Apache HTTP サーバーをインストールできます。

## 4.1. WINDOWS SERVER での APACHE HTTP SERVER アーカイブファイルのダウンロードと展開

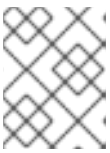
Red Hat カスタマーポータルの ソフトウェアダウンロード ページから、Apache HTTP Server アーカイブファイルをダウンロードできます。ベースとなる JBCS Apache HTTP Server 2.4.57 リリースのアーカイブファイルは、Software Downloads ページの **Releases** タブからダウンロードできます。最新のサービスパックリリースがある場合は、Software Downloads ページの **Security Advisories** タブからダウンロードすることもできます。

**注記**

目的のインストールフォルダーへの書き込みアクセス権を持っている場合は、管理者以外の権限でアーカイブファイルをインストールできます。

**手順**

1. ブラウザーを開き、Red Hat カスタマーポータルの Software Downloads ページにログインします。
2. **Product** ドロップダウンメニューから、**Apache HTTP Server** を選択します。
3. **Version** ドロップダウンメニューから、正しい JBCS バージョンを選択します。
4. **Releases** タブで、**Red Hat JBoss Core Services Apache HTTP Server 2.4.57 for Windows Server x86_64** ファイルの横にある **Download** をクリックします。
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。

   **注記**

   Windows Server システムでは、Apache HTTP Server を **C:\Program Files** ディレクトリーにインストールします。

   アーカイブファイルを展開すると、Apache HTTP Server の最上位フォルダー **jbcs-httpd24-2.4** が自動的に作成されます。このドキュメントでは、**jbcs-httpd24-2.4** フォルダーを **HTTPD_HOME** と呼びます。
6. 最新のサービスパックリリースがある場合は、次の手順を実行してインストールします。
   a. Software Downloads ページで、**Security Advisories** タブをクリックします。
   b. **Security Advisories** タブで、最新の **Red Hat JBoss Core Services Apache HTTP Server 2.4.57 Patch X for Windows Server x86_64** ファイルの横にある **Download** をクリックします。

**注記**

サービスパックのリリースは累積的です。最新のサービスパックリリースをダウンロードすると、以前のサービスパックリリースも自動的にインストールされます。

## 4.2. WINDOWS SERVER での APACHE HTTP サーバーの設定

Windows Server に JBCS Apache HTTP Server をインストールすると、コマンドプロンプトから、または Computer Management Tool を使用して Apache HTTP Server を管理できます。Windows Server 上で Apache HTTP Server を実行する前に、以下の一連の設定作業を行う必要があります。

- Apache HTTP Server のインストール後のスクリプトを実行する
- Apache HTTP Server サービスをインストールする
- Apache HTTP Server サービスのフォルダーパーミッションを設定する
- SSL サポートを無効または有効にする

### 4.2.1. Windows Server での Apache HTTP Server インストール後のスクリプトの実行

Windows Server で Apache HTTP Server を初めて実行する前に、Apache HTTP Server のインストール後のスクリプトを実行する必要があります。

**手順**

1. 管理ユーザーとして **コマンドプロンプト** を開きます。
2. **HTTPD_HOME\etc** ディレクトリーに移動します。
3. 以下のコマンドを入力します。

```
call postinstall.httpd.bat
```

### 4.2.2. Apache HTTP Server サービスのインストール

Windows Server で Apache HTTP Server を初めて実行する前に、Apache HTTP Server を Windows サービスとしてインストールする必要があります。

**注記**

デフォルトでは、Apache HTTP Server はポート 80 を使用するように設定されています。Microsoft Internet Information Services (IIS) がインストールされている場合は、ポートの競合を避けるために、Microsoft IIS を無効にするか、再設定する必要があります。

- **World Wide Web** サービスを停止し、**Startup Type** を **Manual** に変更します。
- IIS を異なるポートを使用するように設定します。

または、Apache HTTP Server サービスをインストールする前に **httpd.conf** を編集し、**Listen** を IIS ポートと競合しないポートに変更することができます。

**前提条件**

- Apache HTTP Server のインストール後のスクリプトを実行 している。

**手順**

1. 管理ユーザーとして **コマンドプロンプト** を開きます。
2. **HTTPD_HOME\bin** ディレクトリーに移動します。
3. 以下のコマンドを使用して、Apache HTTP Server サービスをインストールします。

```
httpd -k install
```

**注記**

ファイアウォールセキュリティーダイアログが表示され、Apache HTTP Server のネットワークアクセスが要求される場合があります。**Allow** をクリックして、ネットワークからこのサービスにアクセスします。

### 4.2.3. Apache HTTP Server サービスのフォルダーパーミッションの設定

Windows Server で Apache HTTP Server を初めて実行する前に、サービスを実行するために使用するアカウントが **HTTPD_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ことを確認してください。

**前提条件**

- Apache HTTP Server サービスがインストール されている。

**手順**

1. **HTTPD_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ーウィンドウで **OK** をクリックして設定を適用します。

### 4.2.4. SSL サポートの無効化または有効化

Apache HTTP Server を実行する前に、SSL 設定ファイルの名前を変更して、SSL サポートを無効または有効にすることを選択できます。Apache HTTP Server は、デフォルトで SSL をサポートします。

**前提条件**

- Apache HTTP Server サービスのフォルダーパーミッションを設定 している。

**手順**

1. **HTTPD_HOME\conf.d** ディレクトリーに移動します。
2. SSL を有効または無効にするには、次のいずれかの手順を実行します。
   - SSL を無効にするには、**ssl.conf** の名前を **ssl.conf.disabled** に変更します。
   - SSL を再度有効にするには、**ssl.conf.disabled** の名前を **ssl.conf** に変更します。

## 4.3. WINDOWS SERVER での APACHE HTTP SERVER の起動

JBCS Apache HTTP Server を Windows Server にインストールする場合は、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 Apache HTTP Server サービスを開始できます。

**前提条件**

- Apache HTTP Server を設定 している。

**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 Apache2.4
    ```

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

## 4.4. WINDOWS SERVER での APACHE HTTP SERVER の停止

Windows Server に JBCS Apache HTTP Server をインストールする場合は、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 Apache HTTP Server サービスを停止できます。

**前提条件**

- Apache HTTP Server を開始 している。

**手順**

- 以下のいずれかの手順を実行します。

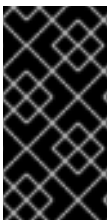
- 管理者としてコマンドプロンプトを開き、以下のコマンドを入力します。

```
net stop Apache2.4
```

- **Start > Administrative Tools > Services**をクリックし、**httpd** サービスを右クリックしてから、**Stop** をクリックします。

# 第5章 APPLICATION STREAMS を使用した RHEL 9 への APACHE HTTP サーバーのインストール

Red Hat Enterprise Linux (RHEL) Application Streams 機能は、**AppStream** リポジトリー内にあるアプリケーション、ランタイム言語、データベースなどのユーザー空間コンポーネントの複数バージョンを配信および更新します。RHEL 9 で、RPM パッケージから Apache HTTP Server をインストールする場合は、Application Streams を使用して Apache HTTP Server の RHEL ディストリビューションをインストールする必要があります。



**重要**

Red Hat JBoss Core Services (JBCS) は、RHEL 9 用の Apache HTTP Server の RPM ディストリビューションを提供しません。RHEL **AppStream** リポジトリーが提供する Apache HTTP Server **httpd** パッケージは、RHEL 9 システムで唯一サポートされる Apache HTTP Server の RPM ディストリビューションです。

**注記**

Apache HTTP Server の RHEL 9 ディストリビューションをインストールしても、**mod_jk** パッケージと **mod_proxy_cluster** パッケージは自動的にインストールされません。RHEL 9 上の RPM パッケージから **mod_jk** および **mod_proxy_cluster** をインストールする方法についての詳細は Apache HTTP Server コネクターおよび負荷分散ガイド を参照してください。

## 5.1. APPLICATION STREMAS を使用する場合の APACHE HTTP サーバーのインストール

標準の **dnf install** コマンドを使用して、RPM パッケージから Apache HTTP Server の RHEL 9 ディストリビューションをインストールできます。その後、root ユーザーとしてコマンドラインから Apache HTTP サーバーを起動および停止できます。システム起動時に Apache HTTP サーバーが自動的に起動するようにすることもできます。

Apache HTTP Server の RHEL ディストリビューションのインストール、起動、停止について、詳しくは Apache HTTP Web サーバーのセットアップ を参照してください。

**関連情報**

- Application Streams (AppStream)
- DNF ツールを使用したソフトウェアの管理

## 5.2. APACHE HTTP SERVER の SELINUX ポリシー

Security-Enhanced Linux (SELinux) ポリシーを使用して、Apache HTTP 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。

Apache HTTP Server の SELinux タイプ名は **httpd_t** です。デフォルトでは、Apache HTTP Server は **/var/www/html** 内のファイルとディレクトリー、および **httpd_sys_content_t** の SELinux タイプコンテキストを持つ他の Web サーバーディレクトリーにアクセスできます。

標準設定以外を使用する場合は、Apache HTTP Server の SELinux ポリシーをカスタマイズすることもできます。

**関連情報**

- SELinux の使用
- 非標準設定での Apache HTTP Server の SELinux ポリシーのカスタマイズ

# 第6章 JBCS APACHE HTTP SERVER の HTTP/2 を有効化する

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。Apache HTTP 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**

Apache HTTP 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。

Multi-Processing Module (MPM) のプリフォーク (**modules/mod_mpm_prefork.so**) を使用している Web サーバーでは、HTTP/2 を利用することはできません。

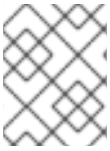
## 6.1. 前提条件

- Red Hat Enterprise Linux の root ユーザーアクセス権を持っている。
- Windows Server で管理者アクセス権を持っている。
- Red Hat JBoss Core Services Apache HTTP Server 2.4.23 以降がインストールされている。
- SSL モジュール (**modules/mod_ssl.so**) がインストールされている。
  SSL モジュールのインストールが必要な場合は、以下のコマンドを入力してください。

```
yum install mod_ssl
```

- HTTP/2 モジュール (**modules/mod_http2.so**) をインストールしている。
  HTTP/2 モジュールをインストールする必要がある場合は、以下のコマンドを入力します。

```
yum install mod_http2
```



**注記**

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。

## 6.2. APACHE HTTP SERVER の HTTP/2 を有効化する

**HTTP_HOME** ディレクトリーにある設定ファイルの設定を更新することで、Apache HTTP Server の HTTP/2 を有効にすることができます。

**手順**

1. **http2_module** を設定に追加する場合は、以下を実行します。

   a. **HTTP_HOME/conf.modules.d/00-base.conf** ファイルを開いてください。

   b. 次の行を入力します。

   ```
   ...
   LoadModule http2_module modules/mod_http2.so
   ```

2. **h2** プロトコルを設定に追加する場合は、以下を実行します。

   a. **HTTP_HOME/conf/httpd.conf** ファイルを開きます。

   b. 仮想ホストの HTTP/2 サポートを有効にするには **h2** プロトコルを仮想ホスト設定に追加します。
   また、すべてのサーバー接続で HTTP/2 サポートを有効にしたい場合は、メインのサーバー設定セクションに **h2** プロトコルを追加します。

   以下に例を示します。

   ```
   <IfModule http2_module>
      Protocols h2 http/1.1
      ProtocolsHonorOrder on
   </IfModule>
   ```

3. Secure Socket Layer (SSL) 設定を更新する場合は、以下を実行します。

   a. **HTTP_HOME/conf.d/ssl.conf** ファイルを開いてください。

   b. **SSLEngine** ディレクティブが enabled に設定されていることを確認します。SSL エンジンはデフォルトで有効になっています。

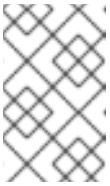
   ```
   SSLEngine on
   ```

   c. **SSLProtocol** ディレクティブを更新し、**SSLv2** および **SSLv3** プロトコルを無効にします。これにより、接続には Transport Layer Security (TLS) が強制的に使用されます。

   ```
   SSLProtocol all -SSLv2 -SSLv3
   ```

   d. **SSLCipherSuite** ディレクティブを更新して、Apache HTTP Server で使用できる SSL 暗号を指定します。
   以下に例を示します。

   ```
   SSLCipherSuite ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-
   SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-
   SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCM-
   SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-
   SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-
   AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-
   SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-
   SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-
   SHA:DHE-RSA-AES256-SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!3DES:!MD5:!PSK
   ```



**注記**

SSL モジュールとサポートされているディレクティブの詳細は、Apache HTTP Server Documentation Version 2.4 - Modules: Apache Module mod_ssl を参照してください。

4. Red Hat JBoss Core Services Apache HTTP Server を再起動し、変更した設定を適用するには、root ユーザーとして次のいずれかの手順を実行します。

   - **systemd** を使用して Red Hat Enterprise Linux で Apache HTTP Server を起動する場合は、次のコマンドを入力します。

     ```
     # systemctl restart jbcs-httpd24-httpd.service
     ```

   - **apachectl** を使用して Red Hat Enterprise Linux で Red Hat JBoss Core Services を起動する場合は、次のコマンドを入力します。

     ```
     # HTTP_HOME/sbin/apachectl restart
     ```

   - Windows Server で Apache HTTP Server を起動する場合は、次のコマンドを入力します。

     ```
     # net restart Apache2.4
     ```

**関連情報**

- HTTP/2 モジュールとサポートされるディレクティブの詳細は、Apache HTTP Server Documentation Version 2.4 - Modules: Apache Module mod_http2 を参照してください。
- SSL モジュールとサポートされているディレクティブの詳細は、Apache HTTP Server Documentation Version 2.4 - Modules: Apache Module mod_ssl を参照してください。

## 6.3. APACHE HTTP SERVER ログを表示して、HTTP/2 が有効化されていることを確認する

Apache HTTP Server のアクセスログまたは要求ログを表示して、HTTP/2 が有効化されていることを確認できます。

**前提条件**

- HTTP/2 を有効化 している。

**手順**

1. ブラウザーから、または **curl** コマンドラインツールを使用して、サーバーにアクセスします。
2. SSL/TLS 要求ログを確認するには、次のコマンドを入力します。

   ```
   $ grep 'HTTP/2' HTTP_HOME/logs/ssl_request_log
   ```

3. SSL/TLS アクセスログを確認するには、次のコマンドを入力します。

   ```
   $ grep 'HTTP/2' HTTP_HOME/logs/ssl_access_log
   ```

**検証**

1. HTTP/2 が有効になっている場合、**grep 'HTTP/2' HTTP_HOME/logs/ssl_request_log** コマンドは次のタイプの出力を生成します。

```
[26/Apr/2018:06:44:45 +0000] 172.17.0.1 TLSv1.2 AES128-SHA "HEAD /html-single/index.html HTTP/2" -
```

2. HTTP/2 が有効になっている場合、**grep 'HTTP/2' HTTP_HOME/logs/ssl_access_log** コマンドは次のタイプの出力を生成します。

```
172.17.0.1 - - [26/Apr/2018:06:44:45 +0000] "HEAD /html-single/index.html HTTP/2" 200 -
```

## 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。

**注記**

Red Hat Enterprise Linux 7 以前で提供される **curl** パッケージは、HTTP/2 をサポートしていません。

**前提条件**

- HTTP/2 を有効化 している。
- **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
```

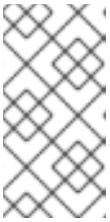
このコマンドは、次のタイプの出力を生成します。

```
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 https://<JBCS_httpd_server>:<port>/<test.html>
```



**注記**

上記の例で、**<JBCS_httpd_server>** を **example.com** などのサーバーの URI に、そして **<test.html>** を設定のテストに使用したい任意の HTML ファイルに置き換えます。HTML のテストページのサンプルは提供されていません。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 HTTP Server Documentation Version 2.4 - How-To / Tutorials: HTTP/2 guide を参照してください。
- SSL 設定の詳細は、Apache HTTP Server Documentation Version 2.4 - SSL/TLS Strong Encryption: How-To 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 OCSP を使用した接続の保護

OCSP (Online Certificate Status Protocol) は、Web ブラウザーおよび Web サーバーがセキュアな接続上で通信できるようにする技術です。暗号化したデータは送信側から送信され、処理前に受信側で復号化されます。Web ブラウザーと Web サーバーは、データの暗号化および復号化を行います。

## 7.1. オンライン証明書ステータスプロトコル

Web ブラウザーと Web サーバーが保護された接続を介して通信する場合、サーバーは認証情報のセットを証明書の形式で提示します。次にブラウザーは証明書を検証し、証明書のステータス情報を求めるリクエストを送信します。サーバーは、現在、期限切れ、または不明の証明書ステータスで応答します。

証明書には、次の種類の情報が含まれています。

- 通信の構文
- Online Certificate Status Protocol (OCSP) レスポンダーにアクセスするための開始時刻、終了時刻、アドレス情報などの制御情報。

Web サーバーは、OCSP レスポンダーを使用して証明書のステータスを確認します。証明書にリストされている OCSP レスポンダーまたは別の OCSP レスポンダーを使用するように Web サーバーを設定できます。OCSP では、期限切れの証明書の猶予期間が許可されます。これにより、証明書更新前の限られた時間内でサーバーにアクセスできます。

OCSP は、証明書失効リスト (CRL) の古いメソッドの制限を解消します。

**関連情報**

- Red Hat Certificate System **計画、インストール、およびデプロイメントのガイド** を参照してください。

## 7.2. SSL 接続用の APACHE HTTP SERVER の設定

**mod_ssl** パッケージをインストールし、**ssl.conf** ファイルで設定を指定することにより、SSL 接続をサポートするように Apache HTTP Server を設定できます。

**前提条件**

- SSL 証明書と秘密鍵を生成しました。
- SSL 証明書と秘密鍵ファイルの場所を知っている。
- SSL 証明書に関連付けられている共通名 (CN) を取得しました。

**手順**

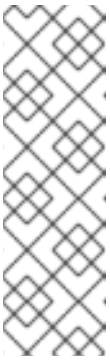
1. **mod_ssl** をインストールするには、次のコマンドを入力します。

```
# yum install jbcs-httpd24-mod_ssl
```

2. SSL 設定を指定するには:
    a. **JBCS_HOME/httpd/conf.d/ssl.conf** ファイルを開きます。

b. **ServerName**、**SSLCertificateFile**、および **SSLCertificateKeyFile** の詳細を入力します。以下に例を示します。

```
<VirtualHost _default_:443>
ServerName www.example.com:443
SSLCertificateFile /opt/rh/jbcs-httpd24/root/etc/pki/tls/certs/localhost.crt
SSLCertificateKeyFile /opt/rh/jbcs-httpd24/root/etc/pki/tls/private/localhost.key
```



**注記**

- **ServerName** は、SSL 証明書に関連付けられている共通名 (CN) と一致する必要があります。**ServerName** が CN に一致しない場合、クライアントブラウザーはドメイン名不一致エラーを表示します。
- **SSLCertificateFile** は、SSL 証明書ファイルへのパスを指定します。
- **SSLCertificateKeyFile** は、SSL 証明書に関連付けられている秘密鍵ファイルへのパスを指定します。

3. **Listen** ディレクティブが、デプロイメントの **httpd** サービスのホスト名または IP アドレスと一致することを確認します。

4. Apache HTTP Server を再起動するには、次のコマンドを入力します。

```
# service jbcs-httpd24-httpd restart
```

## 7.3. APACHE HTTP SERVER での OCSP の使用

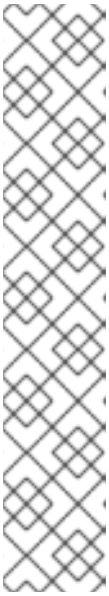
Online Certificate Status Protocol (OCSP) を使用して、Apache HTTP Server との安全な接続を実現できます。

**前提条件**

- SSL 接続用に Apache HTTP Server を設定しました。

**手順**

1. 認証局を設定します。



**注記**

CA が OCSP 証明書を発行できることを確認します。CA は、次の属性を証明書に追加できる必要があります。

```
[ usr_cert ]
...
authorityInfoAccess=OCSP;URI:http://<HOST>:<PORT>
...
[ v3_OCSP ]
basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
extendedKeyUsage = OCSP Signing
```

前の例で、**HOST** と **PORT** を、設定する OCSP レスポンダーの詳細に置き換えます。

2. OCSP レスポンダーを設定します。

**関連情報**

- 証明書と認証局の管理
- OCSP 応答の設定

## 7.4. OCSP 証明書を検証する APACHE HTTP SERVER の設定

**ssl_conf** ファイルで OCSP 設定を定義することにより、OCSP 証明書を検証するように Apache HTTP Server を設定できます。

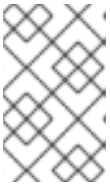
**前提条件**

- 認証局 (CA) を設定しました。
- OCSP レスポンダーを設定しました。

**手順**

1. **JBCS_HOME/httpd/conf.d/ssl.conf** ファイルを開きます。
2. デプロイメントに適した OCSP 設定の詳細を指定します。
以下に例を示します。

```
# Require valid client certificates (mutual auth)
SSLVerifyClient require
SSLVerifyDepth  3
# Enable OCSP
SSLOCSPEnable on
SSLOCSPDefaultResponder http://<HOST>:<PORT>
SSLOCSPOverrideResponder on
```



**注記**

前の例は、クライアント証明書の OCSP 検証を有効にする方法を示しています。前の例で、**<HOST>** と **<PORT>** をデフォルトの OCSP レスポンダーの IP アドレスとポートに置き換えます。

## 7.5. APACHE HTTP SERVER の OCSP 設定の確認

OpenSSL コマンドラインツールを使用して、Apache HTTP Server の OCSP 設定を確認できます。

**手順**

- コマンドラインで、**openssl** コマンドを次の形式で入力します。

```
# openssl ocsp -issuer cacert.crt -cert client.cert -url http://HOST:PORT -CA ocsp_ca.cert
-VAfile ocsp.cert
```

  前述のコマンドで、次の詳細を指定していることを確認してください。

  - **-issuer** オプションを使用して CA 証明書を指定します。
  - **-cert** オプションを使用して、検証するクライアント証明書を指定します。
  - **-url** オプションを使用して、HTTP サーバー検証証明書 (OCSP) を指定します。
  - **-CA** オプションを使用して、Apache HTTP Server サーバー証明書を検証するための CA 証明書を指定します。
  - **-VAfile** オプションを使用して、OCSP レスポンダー証明書を指定します。

**改訂日時:** 2024-02-06