



# Red Hat Process Automation Manager 7.7

IBM WebSphere Application Server への KIE  
Server のインストールおよび設定



# Red Hat Process Automation Manager 7.7 IBM WebSphere Application Server への KIE Server のインストールおよび設定

---

Red Hat Customer Content Services  
brms-docs@redhat.com

## 法律上の通知

Copyright © 2020 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

本書では、KIE Server 用に IBM WebSphere Application Server を設定する方法と、IBM サーバーインスタンスに KIE Server をインストールする方法を説明します。

## 目次

前書き .....	3
第1章 KIE SERVER .....	4
第2章 IBM WEBSPHERE APPLICATION SERVER .....	5
第3章 IBM WEBSPHERE APPLICATION SERVER のインストールおよび実行 .....	6
第4章 KIE SERVER への IBM WEBSPHERE APPLICATION SERVER の設定 .....	8
4.1. 管理セキュリティの有効化 .....	8
4.2. IBM WEBSPHERE APPLICATION SERVER への JDBC データソースの設定 .....	9
4.3. JAVA MESSAGE SERVICE (JMS) の設定 .....	10
4.3.1. サービスバスの作成および IBM WebSphere の追加 .....	11
4.3.2. JMS 接続ファクトリーの作成 .....	11
4.3.2.1. KIE Server の JMS 接続ファクトリー .....	11
4.3.3. JMS キューの作成 .....	12
4.3.3.1. KIE Server 向けの JMS キュー .....	12
4.3.4. JMS アクティベーション仕様の作成 .....	13
4.3.4.1. KIE Server の JMS アクティベーション仕様 .....	13
4.4. IBM WEBSPHERE APPLICATION SERVER へのシステムプロパティの設定 .....	14
4.5. IBM WEBSPHERE APPLICATION SERVER の停止および再起動 .....	17
第5章 IBM WEBSPHERE APPLICATION SERVER への KIE SERVER のインストールおよび実行 .....	19
5.1. KIE SERVER のグループおよびロールの作成 .....	20
5.2. KIE SERVER のグループおよびロールのマッピング .....	20
5.3. KIE SERVER のクラスローディングの設定 .....	21
5.4. インストールの検証 .....	21
第6章 IBM WEBSPHERE APPLICATION SERVER へのヘッドレス PROCESS AUTOMATION MANAGER コント ローラーのインストールおよび実行 .....	23
6.1. ヘッドレス PROCESS AUTOMATION MANAGER コントローラーへのクラスローディングの設定 .....	24
6.2. ヘッドレス PROCESS AUTOMATION MANAGER コントローラーのシステムプロパティの設定 .....	24
6.3. ヘッドレス PROCESS AUTOMATION MANAGER コントローラーグループおよびロールのマッピング .....	26
6.4. インストールの検証 .....	26
第7章 IBM WEBSPHERE APPLICATION SERVER での埋め込みのプロセスエンジンとデシジョンエンジンの設定 .....	28
第8章 キーストアを使用したパスワードセキュリティの確保 .....	33
第9章 次のステップ .....	35
付録A バージョン情報 .....	36



---

## 前書き

システム管理者は、Red Hat KIE Server に IBM WebSphere Application Server を設定し、IBM サーバーインスタンスに KIE Server をインストールできます。

### 前提条件

- IBM WebSphere Application Server インスタンスのバージョン 9.0 以降がインストールされている。詳細なインストール手順は「[IBM WebSphere Application Server 製品ページ](#)」を参照してください。
- WebSphere 統合ソリューションコンソール (通常は <http://<HOST>:9060/ibm/console>) へのアクセスがある。

## 第1章 KIE SERVER

KIE Server は、Red Hat Process Automation Manager のルールおよびその他のアーティファクトの保存先および、実行先のサーバーです。KIE Server はスタンドアロンで組み込みのコンポーネントで、REST、Java Message Service (JMS)、または Java クライアントサイドアプリケーションで利用可能なインターフェースを介してルールをインスタンス化して実行し、ソルバーを介してプロセス、ジョブ、および Red Hat Business Optimizer 機能を管理できます。

Web でデプロイ可能な WAR ファイルとして作成することで、KIE Server は Web コンテナであればどこにでもデプロイできます。KIE Server の現在のバージョンには、Red Hat Decision Manager および Red Hat Process Automation Manager の両方に対するデフォルトの拡張機能が含まれます。

KIE Server はメモリー消費が最小限でフットプリントが小さいため、クラウドインスタンスに簡単にデプロイできます。このサーバーの各インスタンスでは、複数のコンテナを開いてインスタンスを作成できるため、並行して複数のルールサービスを実行できます。

KIE Server は、Oracle WebLogic Server、IBM WebSphere Application Server などのアプリケーションサーバーと統合して、Red Hat Process Automation Manager のアプリケーション管理を合理化できます。



## 第2章 IBM WEBSHERE APPLICATION SERVER

IBM WebSphere Application Server は、Java ベースの Web アプリケーションをホストし、Java EE 認定ランタイム環境を提供する、柔軟性が高くかつ安全な Web アプリケーションです。IBM WebSphere 9.0 は Java SE 8 に対応しており、Java EE 7 に完全に準拠しています。

## 第3章 IBM WEBSHERE APPLICATION SERVER のインストールおよび実行

KIE Server に対応する多くの設定を適用するために、IBM WebSphere Application Server をインストールして実行する必要があります。本セクションは、IBM WebSphere をインストールして起動する方法を説明します。

インストールに関する最新の詳細説明は [IBM Knowledge Center](#) を参照してください。

### 手順

1. IBM Installation Manager バージョン 1.8.5 以降を「[IBM Installation Manager and Packaging Utility download links](#)」ページからダウンロードします。IBM WebSphere のインストールには IBM Installation Manager が必要です。
2. ダウンロードしたアーカイブを展開し、作成されたディレクトリーで、root 権限で以下のコマンドを実行します。

```
sudo ./install
```

IBM Installation Manager が開きます。

3. **File** → **Preferences** の順に移動して、**Add Repository** をクリックします。
4. **Add Repository** ウィンドウに、IBM WebSphere 9.0 のリポジトリー URL を入力します。すべてのリポジトリー URL は、IBM Knowledge Center の「[Online product repositories for IBM WebSphere Application Server offerings](#)」ページで確認できます。
5. 端末で、インストール時に指定した IBM WebSphere Application Server ディレクトリーの場所に移動します。
6. **/bin** ディレクトリーに変更し、以下の例のようなコマンドを実行して、IBM WebSphere のプロファイル、ユーザー名、およびパスワードを作成します。プロファイルは、ランタイム環境を定義します。プロファイルには、ランタイム環境でサーバーが処理し、変更できるファイルがすべて含まれます。このユーザーはログインに必要になります。

```
sudo ./manageprofiles.sh -create -profileName testprofile -profilePath /profiles/testprofile -adminUserName websphere -adminPassword password123
```

7. 端末で、作成したプロファイルの **bin** ディレクトリー (例: **/profiles/testprofile/bin**) に移動し、以下のコマンドを実行して IBM WebSphere Application Server インスタンスを起動します。

```
sudo ./startServer.sh <SERVER_NAME>
```

**<SERVER\_NAME>** は、WebSphere 統合ソリューションコンソールの **Servers** → **Server Types** → **IBM WebSphere Application Servers** で定義した IBM WebSphere Application Server の名前です。

8. Web ブラウザーで、以下の URL を開きます。

```
http://<HOST>:9060/ibm/console
```

**<HOST>** は、ターゲットサーバーの名前または IP アドレスです。

たとえば、システムで起動している IBM WebSphere のローカルインスタンスに対して WebSphere 統合ソリューションコンソールを起動する場合は、以下の URL を Web ブラウザーに入力します。

■ `http://localhost:9060/ibm/console`

9. WebSphere 統合ソリューションコンソールのログインページが表示されたら、管理者の認証情報を入力します。

## 第4章 KIE SERVER への IBM WEBSHERE APPLICATION SERVER の設定

IBM WebSphere Application Server に KIE Server をデプロイする前に、システムプロパティ、セキュリティ設定、JMS 要件、その他の IBM WebSphere プロパティを設定する必要があります。この設定により、KIE Server との最適な統合が促進されます。

### 前提条件

- IBM WebSphere Application Server をインストールして実行している。
- WebSphere 統合ソリューションコンソールにログインしている。

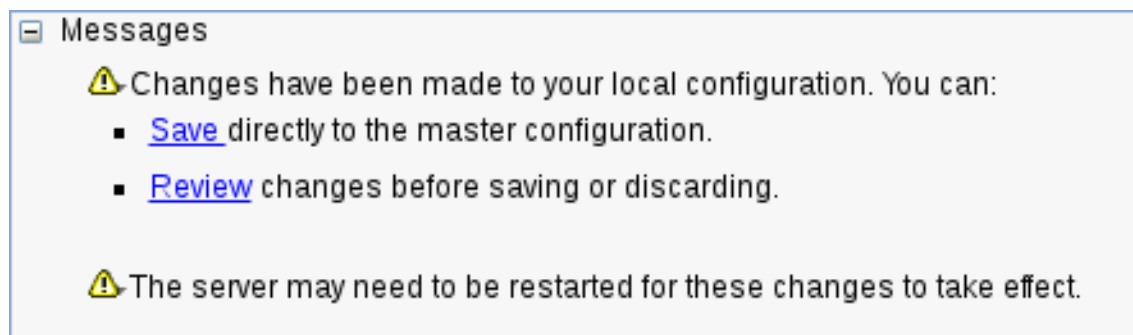
### 4.1. 管理セキュリティの有効化

WebSphere 統合ソリューションコンソールで管理セキュリティを有効にして、ユーザーおよびグループを作成するのに必要な権限を取得します。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Security** → **Global Security** の順にクリックし、**Enable Application Security** オプションが選択されていることを確認します。選択していても、サーバーレベルで上書きされている可能性があります。
2. **Security Configuration Wizard** をクリックし、**Next** をクリックします。
3. ユーザー情報を含むリポジトリを選択します。たとえば、ローカル設定で **Federated repositories** を選択します。
4. **Next** をクリックします。
5. **Primary administrative user name** および **Password** を入力します。
6. **Next** をクリックし、**Finish** をクリックします。
7. **Messages** ウィンドウで **Save** をクリックして、マスター設定への変更を保存します。

図4.1セキュリティ変更の保存



8. 端末で、インストール時に指定した IBM WebSphere Application Server の **/bin** ディレクトリーの場合に移動し、以下のコマンドを実行して IBM WebSphere を停止して再起動し、セキュリティの変更を適用します。

```
sudo ./stopServer.sh <SERVER_NAME>
```

```
sudo ./startServer.sh <SERVER_NAME>
```

<SERVER\_NAME> は、WebSphere 統合ソリューションコンソールの **Servers** → **Server Types** → **IBM WebSphere Application Servers** で定義した IBM WebSphere Application Server の名前です。

## 4.2. IBM WEBSPHERE APPLICATION SERVER への JDBC データソースの設定

データソースは、アプリケーションサーバーなど、Java Database Connectivity (JDBC) クライアントを有効にするオブジェクトで、データベースへの接続を確立します。アプリケーションは、JNDI (Java Naming and Directory Interface) ツリーまたはローカルのアプリケーションコンテキストでデータソースを検索し、データ接続を要求してデータを取得します。IBM WebSphere Application Server にデータソースを設定して、サーバーと、指定したデータベースとの間で適切なデータ交換を行う必要があります。

### 前提条件

- データベース接続を作成するのに使用する JDBC プロバイダーが、データソースをデプロイするすべてのサーバーに設定されている。JDBC プロバイダーの詳細は、IBM Knowledge Center の「[Configuring a JDBC provider](#)」を参照してください。

### 手順

1. Red Hat カスタマーポータルでの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **製品:** Process Automation Manager
  - **バージョン:** 7.7
2. Red Hat Process Automation Manager 7.7.0 Add-Ons をダウンロードします。
3. 以下の手順を実行して、データベースを準備します。
  - a. **TEMP\_DIR** などの一時ディレクトリーに **rhpm-7.7.0-add-ons.zip** を展開します。
  - b. **TEMP\_DIR/rhpm-7.7.0-migration-tool.zip** を展開します。
  - c. 現在のディレクトリーから、**TEMP\_DIR/rhpm-7.7.0-migration-tool/ddl-scripts** ディレクトリーに移動します。このディレクトリーには、複数のデータベースタイプの DDL スクリプトが含まれています。
  - d. 使用するデータベースに、お使いのデータベースタイプの DDL スクリプトをインポートします。以下に例を示します。

```
psql jbpms </ddl-scripts/postgresql/postgresql-jbpms-schema.sql
```

4. WebSphere 統合ソリューションコンソールで、**Resources** → **JDBC** → **Data sources** の順に移動します。
5. データソースを使用するアプリケーションでスコープを選択します。セル、ノード、クラスター、またはサーバーを選択できます。
6. **New** をクリックし、**Create a data source** ウィザードを開きます。

7. **Data source name** フィールドに一意の名前を入力し、**JNDI name** フィールドに JNDI (Java Naming and Directory Interface) 名を入力します。アプリケーションサーバーは JNDI 名を使用して、アプリケーションのリソース参照をこのデータソースにバインドします。複数のリソースタイプ (データソース、J2C 接続ファクトリー/JMS 接続ファクトリー) に割り当てる JNDI 名を重複させないでください。  
同じスコープに、同じタイプの複数のリソースに割り当てる JNDI 名を重複させないでください。
8. プロバイダーを作成している場合は、**Next** をクリックして **Select an existing JDBC provider** を選択します。作成していない場合は **Create new JDBC provider** をクリックして、新しいプロバイダーの詳細を定義します。(JDBC プロバイダーは、データソースを追加するための前提条件であるため、事前に作成しておくことが推奨されます。)
9. CMP (container managed persistence) のエンタープライズ Bean がこのデータソースにアクセスする必要がある場合は、**Enter database specific properties for the data source** パネルで **Use this data source in container managed persistence (CMP)** をクリックします。これにより、その他のデータベース関連のプロパティーが書き込まれます。
10. オプションで、データソースのセキュリティーエイリアスを設定します。すべての認証メソッドで **None** を選択するか、以下のいずれを選択できます。
  - **Component-managed authentication alias:** コンポーネントリソース参照の **res-auth** 値が **Application** の場合に使用する認証エイリアスを指定します。新しいエイリアスを定義するには、**Related Items** → **J2EE Connector Architecture (J2C) authentication data entries** に移動します。コンポーネント管理エイリアスは、データソース認証に対してアプリケーションに指定した ID およびパスワードの組み合わせとなります。したがって、データソースに設定したエイリアスは、アプリケーションコードのエイリアスと同一にする必要があります。
  - **Mapping-configuration alias:** コンポーネントリソース参照にログイン設定がない場合にのみ利用します。**res-auth** 値を **Container** に設定した場合は、ログイン設定およびコンポーネントリソース参照の関連プロパティーを指定することが、認証ストラテジーを定義するのに推奨される方法です。**DefaultPrincipalMapping** ログイン設定を指定する場合の関連プロパティーは JAAS - J2C 認証データエントリーエイリアスです。
  - **Container-managed authentication alias:** コンポーネントリソース参照にログイン設定がない場合にのみ利用します。**res-auth** 値を **Container** に設定する場合は、ログイン設定と、コンポーネントリソース参照の関連プロパティーを指定することで、コンテナ管理認証ストラテジーが判断されます。
11. **Next** をクリックしてデータソースの情報を確認し、**Finish** をクリックして設定を保存してウィザードを終了します。  
**Data sources** パネルには、新しい設定と、同じスコープに設定したその他のデータソースがテーブルに表示されます。

IBM WebSphere Application Server データソースの詳細は、IBM Knowledge Center の「[Configuring a JDBC provider and data source](#)」を参照してください。

### 4.3. JAVA MESSAGE SERVICE (JMS) の設定

Java Message Service (JMS) は、KIE Server が、Oracle WebLogic Server、IBM WebSphere Application Server などのアプリケーションサーバーとメッセージを交換するために使用する Java API です。KIE Server を経由して JMS メッセージを送受信するようにアプリケーションサーバーを設定し、2 台のサーバー間でコラボレーションが適切に行われるようにします。

### 4.3.1. サービスバスの作成および IBM WebSphere の追加

JMS を使用するためには、サービスバスを作成し、IBM WebSphere Application Server をそのメンバーとして追加します。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Service Integration** → **Buses** → **New** の順に移動します。
2. 新しいバス名を入力し、**Bus Security** オプションの選択を解除します。
3. **Next** をクリックしてから **Finish** をクリックし、サービスバスを作成します。
4. 作成しておいたサービスバスを選択します。
5. **Topology** で **Bus Members** → **Add** をクリックします。
6. **Add a New Bus Member** ウィザードを使用し、IBM WebSphere Application Server と、永続性に関するメッセージストアのタイプを選択します。メッセージストアのプロパティを指定することもできます。
7. **Finish** をクリックし、バスメンバーを追加します。

### 4.3.2. JMS 接続ファクトリーの作成

KIE Server でメッセージングを有効にするには、メッセージの送受信に JMS 接続ファクトリーをいくつか作成する必要があります。

#### 前提条件

- IBM WebSphere Application Server にサービスバスを作成している。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Resources** → **JMS** → **Connection Factories** の順に移動します。
2. 正しいスコープを選択し、**New** をクリックします。
3. **Default Messaging Provider** オプションを選択し、**OK** をクリックします。
4. 以下に示す必須の接続ファクトリーに、接続ファクトリーの名前 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/cf/KIE.SERVER.REQUEST**) を入力して、**Bus Name** ドロップダウンリストからサービスバス名を選択します。残りのオプションはデフォルト値のままにします。
5. **Apply** および **Save** をクリックして master 設定への変更を保存し、必要な各ファクトリーに上述の手順を繰り返します。

#### 4.3.2.1. KIE Server の JMS 接続ファクトリー

以下は、KIE Server で JMS メッセージングを有効にするために必要な Java Message Service (JMS) 接続ファクトリーになります。

#### 表4.1 KIE Server に必要な JMS 接続ファクトリー

名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/cf/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信
<b>KIE.SERVER.RESPONSE</b>	<b>jms/cf/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信
<b>KIE.SERVER.EXECUTOR</b>	<b>jms/cf/KIE.SERVER.EXECUTOR</b>	KIE Server エグゼキューターサービス

### 4.3.3. JMS キューの作成

JMS キューは、ポイントツーポイントメッセージング (point-to-point messaging) の宛先エンドポイントになります。KIE Server で JMS メッセージングを有効にするには JMS キューをいくつか作成する必要があります。

#### 前提条件

- IBM WebSphere Application Server にサービスバスを作成している。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Resources** → **JMS** → **Queues** の順に移動します。
2. 正しいスコープを選択し、**New** をクリックします。
3. **Default Messaging Provider** オプションを選択し、**OK** をクリックします。
4. 以下に示す必須のキューに、キューの名前 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/KIE.SERVER.REQUEST**) を入力して、**Bus Name** ドロップダウンリストからサービスバスを選択します。
5. **Queue Name** ドロップダウンリストで **Create Service Integration Bus Destination** を選択して一意の識別子を入力し、事前に作成したバスメンバーを選択します。
6. **Apply** および **Save** をクリックして master 設定への変更を保存し、必要な各キューに上述の手順を繰り返します。

#### 4.3.3.1. KIE Server 向けの JMS キュー

以下は、KIE Server で JMS メッセージングを有効にするために必要な Java Message Service (JMS) キューです。

表4.2 KIE Server に必要な JMS キュー

名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信



名前	デフォルト値	用途
<b>KIE.SERVER.RESPONSE</b>	<b>jms/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信
<b>KIE.SERVER.EXECUTOR</b>	<b>jms/KIE.SERVER.EXECUTOR</b>	KIE Server エグゼキューターサービス

#### 4.3.4. JMS アクティベーション仕様の作成

キューと、JMS を有効にするメッセージ駆動型 Bean との間にブリッジを行うには、JMS アクティベーション仕様が必要です。

##### 前提条件

- IBM WebSphere Application Server にサービスバスを作成している。
- JMS キューを作成している。

##### 手順

1. WebSphere 統合ソリューションコンソールで、**Resources** → **JMS** → **Activation Specifications** の順に移動します。
2. 正しいスコープを選択し、**New** をクリックします。
3. **Default Messaging Provider** オプションを選択し、**OK** をクリックします。
4. 以下に示す必須のアクティベーション仕様に、アクティベーション仕様名 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/activation/KIE.SERVER.REQUEST**) を入力して、**Bus Name** ドロップダウンリストからサービスバス名を選択します。
5. **Destination Type** ドロップダウンリストで **Queue** を選択し、**Destination lookup** に対応するキューの名前 (例: **jms/KIE.SERVER.REQUEST**) を入力します。
6. **Apply** および **Save** をクリックして master 設定への変更を保存し、必要な各アクティベーション仕様に上述の手順を繰り返します。

##### 4.3.4.1. KIE Server の JMS アクティベーション仕様

以下は、KIE Server で JMS メッセージングを有効にするために必要な Java Message Service (JMS) アクティベーション仕様になります。

表4.3 KIE Server に必要な JMS アクティベーション仕様

名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/activation/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信

名前	デフォルト値	用途
<b>KIE.SERVER.RESPONSE</b>	<b>jms/activation/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信
<b>KIE.SERVER.EXECUTOR</b>	<b>jms/activation/KIE.SERVER.EXECUTOR</b>	KIE Server エグゼキューターサービス

## 4.4. IBM WEBSHERE APPLICATION SERVER へのシステムプロパティの設定

KIE Server をデプロイする前に、IBM WebSphere Application Server に、本セクションに記載するシステムプロパティを設定します。

### 手順

1. WebSphere 統合ソリューションコンソールで、**Servers** → **Server Types** → **IBM WebSphere Application Servers** の順にクリックします。
2. アプリケーションサーバーの一覧で、KIE Server をデプロイするサーバーを選択します。
3. **Server Infrastructure** で **Java and Process Management** → **Process Definition** をクリックします。

図4.2 WebSphere 設定ページ

The screenshot shows the 'Application servers' configuration page for 'server1'. The page is divided into several sections:

- General Properties:** Includes fields for 'Name' (server1) and 'Node name' (dhcp-4-116Node01). There are checkboxes for 'Run in development mode', 'Parallel start' (checked), and 'Start components as needed'. A dropdown menu for 'Access to internal server classes' is set to 'Allow'.
- Server-specific Application Settings:** Includes a dropdown for 'Classloader policy' (Multiple) and a dropdown for 'Class loading mode' (Classes loaded with parent class loader first).
- Container Settings:** A list of expandable settings including Session management, SIP Container Settings, Web Container Settings, Portlet Container Settings, EJB Container Settings, Container Services, and Business Process Services.
- Applications:** A link for 'Installed applications'.
- Server messaging:** A list of links for Messaging engines, Messaging engine inbound transports, WebSphere MQ link inbound transports, and SIB service.
- Server Infrastructure:** A list of expandable settings including Java and Process Management (with sub-items Class loader, Process definition, and Process execution) and Administration (with sub-item Java SDKs).

At the bottom of the configuration area, there are buttons for 'Apply', 'OK', 'Reset', and 'Cancel'.

4. **Additional Properties** で **Java Virtual Machine** をクリックします。

図4.3 プロセスの定義設定ページ

これにより、IBM WebSphere を起動する際に使用する JVM の設定プロパティが開きます。

5. **Initial heap size** および **Maximum heap size** の両方を **2048** に設定し、**Apply** をクリックして Java Virtual Machine (JVM) メモリーサイズを増やします。KIE Server はこの値でテストされています。JVM メモリーサイズを増やさないと、KIE Server のデプロイ時に IBM WebSphere Application Server がフリーズするかエラーが発生します。
6. **Additional Properties** で **Custom Properties** をクリックします。
7. **New** → **Custom JVM Properties** の順にクリックし、以下のプロパティを IBM WebSphere に追加します。

表4.4 KIE Server のシステムプロパティ

名前	値	説明
<b>kie.server.jms.queues.response</b>	<b>jms/KIE.SERVER.RESPONSE</b>	KIE Server が使用する応答用の JMS キューの JNDI 名。
<b>org.kie.server.domain</b>	<b>WSLogin</b>	JMS の使用時にユーザーの認証に使用される JAAS <b>LoginContext</b> ドメイン。

名前	値	説明
<code>org.kie.server.persistence.ds</code>	<code>jdbc/jbpm</code>	KIE Server のデータソースの JNDI 名。
<code>org.kie.server.persistence.tm</code>	<code>org.hibernate.engine.transaction.jta.platform.internal.WebSphereJtaPlatform</code>	Hibernate プロパティを設定するためのトランザクションマネージャープラットフォーム。
<code>org.kie.server.persistence.dialect</code>	例: <code>org.hibernate.dialect.H2Dialect</code>	使用する Hibernate 方言を指定します。データソースに従って設定します。
<code>org.kie.executor.jms.queue</code>	<code>jms/KIE.SERVER.EXECUTOR</code>	KIE Server へのジョブエグゼキューターの JMS キュー。
<code>org.kie.executor.jms.cf</code>	<code>jms/cf/KIE.SERVER.EXECUTOR</code>	KIE Server へのジョブエグゼキューターの JMS 接続ファクトリー。
<code>org.kie.server.router</code>	例: <code>http://localhost:9000</code>	(オプション) クラスター化された KIE Server 環境で、アプリケーションサーバーがその一部である1つ以上の KIE Server ルーター (Smart Router) の URL を1つ以上指定します。
<code>org.jboss.logging.provider</code>	<code>jdk</code>	このプロパティは、 <b>CA SiteMinder TAI (SMTAI)</b> が環境にインストールされている場合にのみ必要になります。このプロパティを使用すると、Dashbuilder でログを記録するために、Hibernate が <b>log4j</b> ではなく、 <b>JDK</b> を強制的に使用しようとします。 <b>CA SiteMinder TAI (SMTAI)</b> には古いバージョンの <b>log4j</b> が含まれており、これにより競合が生じます。

8. **Save** をクリックして、マスター設定への変更を保存します。

#### 4.5. IBM WEBSHERE APPLICATION SERVER の停止および再起動

IBM WebSphere Application Server に必要なシステムプロパティをすべて設定したあと、IBM サーバーを停止および再起動して、設定が適用されていることを確認します。

## 手順

端末で、インストール時に指定した IBM WebSphere Application Server の **/bin** ディレクトリーの場所に移動し、以下のコマンドを実行して IBM WebSphere を停止して再起動し、設定変更を適用します。

```
sudo ./stopServer.sh <SERVER_NAME>
```

```
sudo ./startServer.sh <SERVER_NAME>
```

**<SERVER\_NAME>** は、WebSphere 統合ソリューションコンソールの **Servers** → **Server Types** → **IBM WebSphere Application Servers** で定義した IBM WebSphere Application Server の名前です。

## 第5章 IBM WEBSHERE APPLICATION SERVER への KIE SERVER のインストールおよび実行

IBM WebSphere Application Server に必要なシステムプロパティをすべて設定したら、IBM WebSphere に KIE Server をインストールして、Red Hat Process Automation Manager アプリケーション管理を合理化します。

### 前提条件

- 「[4章 KIE Server への IBM WebSphere Application Server の設定](#)」の説明通りに IBM WebSphere Application Server インスタンスを設定している。

### 手順

- Red Hat カスタマーポータルでの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - 製品: Process Automation Manager
  - バージョン: 7.7
- Red Hat Process Automation Manager 7.7.0 KIE Server for All Supported EE7 Containers をダウンロードします。
- rhpm-7.7.0-kie-server-ee7.zip** アーカイブを一時ディレクトリーに展開します。以下の例では、このディレクトリーを **TEMP\_DIR** とします。
- kie-server.war** ディレクトリーを再パッケージ化します。
  - TEMP\_DIR/rhpm-7.7.0-kie-server-ee7/kie-server.war** ディレクトリーに移動します。
  - TEMP\_DIR/rhpm-7.7.0-kie-server-ee7/kie-server.war** ディレクトリーのコンテンツを選択し、**kie-server.zip** ファイルを作成します。
  - kie-server.zip** の名前を **kie-server.war** に変更します。このファイルを使用して、KIE Server をデプロイします。
  - オプションで、新しい **kie-server.war** ファイルを、デプロイしやすい便利な場所にコピーします。
- WebSphere 統合ソリューションコンソールで、**Applications → Application Types → WebSphere Enterprise Applications** の順に移動します。
- Install** をクリックします。
- 再パッケージ化した **kie-server.war** ファイルに移動して選択し、アップロードします。
- Fast Path** を選択し、**Next** をクリックします。**Install New Application** ウィザードが開きます。
- Application Name** を **kie-server** に変更し、**Next** をクリックします。
- 要件に合わせて KIE Server モジュールをサーバーにマッピングし、**Next** をクリックします。
- Bind Listeners for Message-Driven Beans** の場合は、両 Bean に **Activation Specification** を選択し、**Target Resource JNDI Name** フィールドに **jms/activation/KIE.SERVER.REQUEST**

を入力し、**KIE.SERVER.REQUEST** 接続ファクトリーの **jms/cf/KIE.SERVER.REQUEST** JNDI 名を入力します。

12. **Map Virtual Hosts for Web Modules** セクションでは、デフォルト値をそのままにして、**Next** をクリックします。
13. コンテキストのルートを **kie-server** に設定します。
14. **Metadata for Modules** セクションで、デフォルト値をそのままにして **Next** をクリックします。
15. **Finish** をクリックして KIE Server をインストールし、**Save** をクリックして master 設定に変更を保存します。

## 5.1. KIE SERVER のグループおよびロールの作成

KIE Server をインストールしたら、**kie-server** グループと、ユーザーを作成します。

### 前提条件

- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。

### 手順

1. WebSphere 統合ソリューションコンソールで、**Users and Groups** → **Manage Groups** の順にクリックします。
2. **Manage Groups** 画面で **Create** をクリックします。
3. **Create a Group** 画面の **Group name** ボックスに **kie-server** と入力し、**Create** をクリックします。
4. **kie-server** グループに追加するユーザーを作成するには、**Users and Groups** → **Manage Users** の順にクリックします。
5. **Create a User** セクションで、必要な情報を入力します。
6. **Group Membership** をクリックします。
7. **Group Membership** 画面で **kie-server** をクリックして **Mapped To** に移動し、**Close** をクリックします。
8. **Create a User** 画面で **Create** をクリックします。

## 5.2. KIE SERVER のグループおよびロールのマッピング

KIE Server をインストールしたら、WebSphere 統合ソリューションコンソールで **kie-server** ロールを **kie-server** グループにマッピングして、KIE Server を実行します。

### 前提条件

- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。
- IBM WebSphere Application Server には、1人以上のユーザーが追加されている **kie-server** グループがある。



## 手順

1. WebSphere 統合ソリューションコンソールで、**Applications** → **Application Types** → **WebSphere Enterprise Applications** に移動し、新たにインストールした **kie-server** アプリケーションを選択します。
2. **Detail Properties** で **Security Role to User/Group Mapping** をクリックします。
3. **kie-server** ロールを選択し、**Map Groups** をクリックして、**kie-server** グループを検索します。
4. **kie-server** グループを、**Available** 一覧から **Selected** 一覧に移動し、**OK** をクリックします。このマッピングにより、IBM WebSphere Application Server の **kie-server** グループのユーザーに、KIE Server へのアクセスが付与されます。
5. **Save** をクリックしてマッピングを完了します。

## 5.3. KIE SERVER のクラスローディングの設定

KIE Server をインストールしたら、親クラスを最後にロードするように、クラスローディングを設定する必要があります。

## 手順

1. **Applications** → **Application Types** → **WebSphere Enterprise Applications** に移動し、**kie-server** をクリックします。
2. 左側の **Detail Properties** の見出しにある **Class Loading and Update Detection** をクリックします。
3. プロパティーで、**Class Loader Order** を **Classes loaded with local class loader first (parent last)** に、**WAR Class Loader Policy** を **Single class loader for application** に変更します。
4. 変更をマスター設定に保存します。

## 5.4. インストールの検証

KIE Server をインストールして KIE Server グループマッピングを定義したら、サーバーが実行していることを確認します。

## 前提条件

- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。
- ヘッドレス Process Automation Manager コントローラーに必要なシステムプロパティーをすべて設定している。
- IBM WebSphere Application Server に KIE Server グループマッピングを定義している。

## 手順

KIE Server の URL (**http://<HOST>:<PORT>/kie-server**) に移動し、サーバーが実行していることを確認するか、**GET** 要求を **http://<HOST>:<PORT>/kie-server/services/rest/server** に送信して、KIE Server REST API が応答するかどうかを確認します。

<HOST> は、KIE Server ホストの ID または名前です (例: **localhost** または **192.7.8.9**)。

<**PORT**> は、KIE Server ホストのポートです (例: **9060**)。

KIE Server が実行していない場合は、IBM WebSphere Application Server インスタンスを停止して再起動し、KIE Server の URL または API に再度アクセスしてみてください。

## 第6章 IBM WEBSHERE APPLICATION SERVER へのヘッドレス PROCESS AUTOMATION MANAGER コントローラーのインストールおよび実行

KIE Server REST API または Java Client API を使用して KIE Server に接続するには、IBM WebSphere Application Server にヘッドレス Process Automation Manager コントローラーをインストールします。ヘッドレス Process Automation Manager コントローラーは、一元化された方法で KIE Server 設定を管理するため、このコントローラーを使用してコンテナの作成および維持を行い、サーバーレベルのタスクを実行できます。



### 注記

実稼働環境で最適な結果を得るには、KIE Server とヘッドレス Process Automation Manager コントローラーを別のサーバーにインストールします。開発環境の場合は、KIE Server とヘッドレス Process Automation Manager コントローラーを同じサーバーにインストールできます。

### 前提条件

- 「[4章 KIE Server への IBM WebSphere Application Server の設定](#)」の説明通りに IBM WebSphere Application Server インスタンスを設定している。
- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。
- インストールを完了するのに必要なユーザーパーミッションがある。

### 手順

1. Red Hat カスタマーポータルでの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **製品:** Process Automation Manager
  - **バージョン:** 7.7
2. Red Hat Process Automation Manager 7.7.0 Add-Ons をダウンロードします。
3. ダウンロードした **rhpmam-7.7.0-add-ons.zip** ファイルを一時ディレクトリーに展開します。
4. **rhpmam-7.7.0-add-ons.zip** ファイルを一時ディレクトリーに展開します。以下の例では、このディレクトリーを **TEMP\_DIR** とします。
5. **rhpmam-7.7.0-add-ons/rhpmam-7.7.0-controller-ee7.zip** ファイルを展開します。
6. **controller.war** ディレクトリーを再パッケージ化します。
  - a. **TEMP\_DIR/rhpmam-7.7.0-add-ons/rhpmam-7.7.0-controller-ee7/controller.war** ディレクトリーに移動します。
  - b. **TEMP\_DIR/rhpmam-7.7.0-add-ons/rhpmam-7.7.0-controller-ee7/controller.war** ディレクトリーの内容を選択して、**controller.zip** ファイルを作成します。
  - c. **controller.zip** の名前を **controller.war** に変更します。これは、ヘッドレス Process Automation Manager コントローラーをデプロイするのに使用するファイルです。

- d. オプションで、新しい **controller.war** ファイルを、デプロイしやすい場所にコピーします。
7. WebSphere 統合ソリューションコンソールで、**Applications** → **Application Types** → **WebSphere Enterprise Applications** の順に移動します。
8. **Install** をクリックします。
9. 再パッケージ化した **controller.war** ファイルに移動し、選択してアップロードします。
10. **Fast Path** を選択し、**Next** をクリックします。  
**Install New Application** ウィザードが開きます。
11. **Application Name** を **controller** に変更し、**Next** をクリックします。
12. 要件に合わせてヘッドレス Process Automation Manager コントローラーモジュールをサーバーにマッピングし、**Next** をクリックします。
13. **Bind Listeners for Message-Driven Beans** の場合は、両 Bean に **Activation Specification** を選択し、**Target Resource JNDI Name** フィールドに **jms/activation/KIE.SERVER.REQUEST** を入力し、**KIE.SERVER.REQUEST** 接続ファクトリーの **jms/cf/KIE.SERVER.REQUEST** JNDI 名を入力します。
14. **Map Virtual Hosts for Web Modules** セクションでは、デフォルト値をそのままにして、**Next** をクリックします。
15. コンテキストルートを **controller** に設定します。
16. **Metadata for Modules** セクションで、デフォルト値をそのままにして **Next** をクリックします。
17. **Finish** をクリックしてヘッドレス Process Automation Manager コントローラーをインストールし、**Save** をクリックして master 設定への変更を保存します。

## 6.1. ヘッドレス PROCESS AUTOMATION MANAGER コントローラーへのクラスローディングの設定

ヘッドレス Process Automation Manager コントローラーをインストールしたら、親クラスを最後にロードするように、KIE Server クラスローディングを設定する必要があります。

### 手順

1. **Applications** → **Application Types** → **WebSphere Enterprise Applications** に移動し、**kie-server** をクリックします。
2. 左側の **Detail Properties** の見出しにある **Class Loading and Update Detection** をクリックします。
3. プロパティーで、**Class Loader Order** を **Classes loaded with local class loader first (parent last)** に、**WAR Class Loader Policy** を **Single class loader for application** に変更します。
4. 変更をマスター設定に保存します。

## 6.2. ヘッドレス PROCESS AUTOMATION MANAGER コントローラーのシステムプロパティーの設定

ヘッドレス Process Automation Manager コントローラーをインストールしたら、本セクションで紹介するシステムプロパティをアプリケーションサーバーに設定して、ヘッドレス Process Automation Manager コントローラーとの適切な接続を有効にします。



## 注記

実稼働環境で最適な結果を得るには、KIE Server とヘッドレス Process Automation Manager コントローラーを別のサーバーにインストールします。開発環境の場合は、KIE Server とヘッドレス Process Automation Manager コントローラーを同じサーバーにインストールします。いずれの場合も、ヘッドレス Process Automation Manager コントローラーがインストールされているすべてのアプリケーションサーバーでこのプロパティを変更します。

## 前提条件

- KIE Server およびヘッドレス Process Automation Manager コントローラーがアプリケーションサーバーインスタンスにインストールされている。

## 手順

1. ヘッドレス Process Automation Manager コントローラーがインストールされているアプリケーションサーバーインスタンスに、以下の JVM プロパティ値を指定します。

表6.1 ヘッドレス Process Automation Manager コントローラーに必要なプロパティ

名前	要件
<b>org.kie.server.user</b>	<b>kie-server</b> ロールが割り当てられているユーザー
<b>org.kie.server.pwd</b>	<b>org.kie.server.user</b> プロパティに指定したユーザーのパスワード

2. KIE Server がインストールされているアプリケーションサーバーインスタンスに、以下の JVM プロパティ値を指定します。

表6.2 ヘッドレス Process Automation Manager コントローラーがインストールされている場合に KIE Server に必要なプロパティ

名前	要件
<b>org.kie.server.controller.user</b>	<b>kie-server</b> ロールが割り当てられているユーザー
<b>org.kie.server.controller.pwd</b>	<b>org.kie.server.controller.user</b> プロパティに指定したユーザーのパスワード
<b>org.kie.server.id</b>	KIE Server インストールの ID または名前 ( <b>rhdm700-decision-server-1</b> など)
<b>org.kie.server.location</b>	KIE Server の URL ( <b>http://&lt;HOST&gt;:&lt;PORT&gt;/kie-server/services/rest/server</b> )

名前	要件
<b>org.kie.server.controller</b>	ヘッドレス Process Automation Manager コントローラーの URL ( <b>http://&lt;HOST&gt;:&lt;PORT&gt;/controller/rest/controller</b> )

<HOST> は、KIE Server ホストの ID または名前です (例: **localhost** または **192.7.8.9**)。

<PORT> は、KIE Server ホストのポートです (例: **9060**)。

### 6.3. ヘッドレス PROCESS AUTOMATION MANAGER コントローラーグループおよびロールのマッピング

ヘッドレス Process Automation Manager コントローラーをインストールしたら、WebSphere 統合ソリューションコンソールに **kie-server** グループを作成し、**kie-server** ロールを **kie-server** グループにマッピングします。

#### 前提条件

- ヘッドレス Process Automation Manager コントローラーが IBM WebSphere Application Server インスタンスにインストールされている。

#### 手順

1. WebSphere 統合ソリューションコンソールで、「[KIE Server グループおよびロールの作成](#)」の記載通りに、**kie-server** グループと、このグループのユーザーを作成します。
2. **Applications** → **Application Types** → **WebSphere Enterprise Applications** に移動し、新たにインストールしたヘッドレス Process Automation Manager コントローラーを選択します。
3. **Detail Properties** で **Security Role to User/Group Mapping** をクリックします。
4. **kie-server** ロールを選択し、**Map Groups** をクリックして、**kie-server** グループを検索します。
5. **kie-server** グループを、**Available** 一覧から **Selected** 一覧に移動し、**OK** をクリックします。このマッピングにより、事前に作成した管理者ユーザーに Process Automation Manager へのアクセスが付与されます。
6. **Save** をクリックしてマッピングを完了します。

### 6.4. インストールの検証

ヘッドレス Process Automation Manager コントローラーをインストールし、必要なシステムプロパティとロール要件をアプリケーションサーバーに定義したら、ヘッドレス Process Automation Manager コントローラーが正しく動作していることを確認します。

#### 前提条件

- KIE Server およびヘッドレス Process Automation Manager コントローラーがアプリケーションサーバーインスタンスにインストールされている。
- アプリケーションサーバーで、ヘッドレス Process Automation Manager コントローラーに必要なシステムプロパティおよびロール要件をすべて設定している。

## 手順

端末で以下のコマンドを実行して、ヘッドレス Process Automation Manager コントローラーが動作していることを確認します。

```
curl -X GET "http://<HOST>:<PORT>/controller/rest/controller/management/servers" -H "accept: application/xml" -u '<CONTROLLER>:<CONTROLLER_PWD>'
```

<HOST> は、KIE Server ホストの ID または名前です (例: **localhost** または **192.7.8.9**)。

<PORT> は、KIE Server ホストのポートです (例: **9060**)。

<CONTROLLER> および <CONTROLLER\_PWD> は、本セクションで作成したユーザーの認証情報です。

このコマンドにより、KIE Server インスタンスに関する情報が返されます。



### 注記

あるいは、KIE Server Java API Client を使用して、ヘッドレス Process Automation Manager コントローラーにアクセスすることもできます。

ヘッドレス Process Automation Manager コントローラーが実行していない場合は、アプリケーションサーバーインスタンスを停止して再起動し、ヘッドレス Process Automation Manager コントローラーの URL または API へ再度アクセスしてみてください。

## 第7章 IBM WEBSHERE APPLICATION SERVER での埋め込みのプロセスエンジンとデシジョンエンジンの設定

埋め込みエンジンとは、デシジョンとビジネスプロセスを実行できるようにする軽量のワークフローおよびルールエンジンです。埋め込みエンジンは、Red Hat Process Automation Manager アプリケーションに含めるか、OpenShift、Kubernetes および Docker 経由でサービスとしてデプロイできます。また、API を使用して Red Hat Process Automation Manager アプリケーションにエンジンを埋め込むか、または contexts and dependency injection (CDI) サービスセットとして、埋め込むことができます。

Red Hat Process Automation Manager アプリケーションで埋め込みされたエンジンを使用する予定の場合には、Red Hat Business Automation 部品表 (BOM) ファイルをプロジェクトの **pom.xml** ファイルに追加して、プロジェクトに、Maven の依存関係を追加する必要があります。Red Hat Business Automation BOM は、Red Hat Decision Manager および Red Hat Process Automation Manager に適用されます。Red Hat Business Automation BOM に関する詳細は、「[What is the mapping between Red Hat Process Automation Manager and the Maven library version?](#)」を参照してください。

### 手順

1. Red Hat Business Automation BOM を **pom.xml** ファイルで宣言します。

```
<dependencyManagement>
<dependencies>
<dependency>
<groupId>com.redhat.ba</groupId>
<artifactId>ba-platform-bom</artifactId>
<version>7.7.0.redhat-00002</version>
<type>pom</type>
<scope>import</scope>
</dependency>
</dependencies>
</dependencyManagement>
<dependencies>
<!-- Your dependencies -->
</dependencies>
```

2. **<dependencies>** タグでお使いのプロジェクトに必要な依存関係を宣言します。製品の BOM をプロジェクトにインポートしたら、ユーザー向け製品依存関係のバージョンが定義されるため、**<dependency>** 要素のサブ要素 **<version>** を指定する必要はありません。ただし、**<dependency>** 要素を使用して、プロジェクトで使用する依存関係を宣言する必要があります。
  - 標準的な Red Hat Process Automation Manager プロジェクトでは、使用する機能に応じて、以下の依存関係を宣言します。

### 埋め込みプロセスエンジン依存関係

```
<!-- Public KIE API -->
<dependency>
<groupId>org.kie</groupId>
<artifactId>kie-api</artifactId>
</dependency>

<!-- Core dependencies for process engine -->
<dependency>
```



```

    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-flow</artifactId>
  </dependency>

  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-flow-builder</artifactId>
  </dependency>

  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-bpmn2</artifactId>
  </dependency>

  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-runtime-manager</artifactId>
  </dependency>

  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-persistence-jpa</artifactId>
  </dependency>

  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-query-jpa</artifactId>
  </dependency>

  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-audit</artifactId>
  </dependency>

  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-kie-services</artifactId>
  </dependency>

  <!-- Dependency needed for default WorkItemHandler implementations. -->
  <dependency>
    <groupId>org.jbpm</groupId>
    <artifactId>jbpm-workitems-core</artifactId>
  </dependency>

  <!-- Logging dependency. You can use any logging framework compatible with slf4j. -->
  <dependency>
    <groupId>ch.qos.logback</groupId>
    <artifactId>logback-classic</artifactId>
    <version>${logback.version}</version>
  </dependency>

```

- CDI を使用する Red Hat Process Automation Manager プロジェクトでは、通常、以下の依存関係を宣言します。

### CDI が有効化されたプロセスエンジンの依存関係

```

<dependency>
  <groupId>org.kie</groupId>
  <artifactId>kie-api</artifactId>
</dependency>

<dependency>
  <groupId>org.jbpm</groupId>
  <artifactId>jbpm-kie-services</artifactId>
</dependency>

<dependency>
  <groupId>org.jbpm</groupId>
  <artifactId>jbpm-services-cdi</artifactId>
</dependency>

```

### 埋め込みデシジョンエンジン依存関係

```

<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-compiler</artifactId>
</dependency>

<!-- Dependency for persistence support. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-persistence-jpa</artifactId>
</dependency>

<!-- Dependencies for decision tables, templates, and scorecards.
For other assets, declare org.drools:business-central-models-* dependencies. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-decisiontables</artifactId>
</dependency>
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-templates</artifactId>
</dependency>
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-scorecards</artifactId>
</dependency>

<!-- Dependency for loading KJARs from a Maven repository using KieScanner. -->
<dependency>
  <groupId>org.kie</groupId>
  <artifactId>kie-ci</artifactId>
</dependency>

```

- KIE Server を使用するには、以下の依存関係を宣言します。

### クライアントアプリケーション KIE Server の依存関係

```

<dependency>
  <groupId>org.kie.server</groupId>

```

```
<artifactId>kie-server-client</artifactId>
</dependency>
```

- Red Hat Process Automation Manager にリモートクライアントを作成するには、以下の依存関係を宣言します。

### クライアントの依存関係

```
<dependency>
  <groupId>org.uberfire</groupId>
  <artifactId>uberfire-rest-client</artifactId>
</dependency>
```

- ルール、プロセス定義など、アセットを含む JAR ファイルを作成する場合は、お使いの Maven プロジェクトのパッケージングの種類を **kjar** とし、**org.kie:kie-maven-plugin** を使用して、**<project>** 要素に置かれた **kjar** パッケージングタイプを処理します。以下の例の **\${kie.version}** は、「[What is the mapping between Red Hat Process Automation Manager and the Maven library version?](#)」に記載されている Maven ライブラリーのバージョンです。

```
<packaging>kjar</packaging>
<build>
  <plugins>
    <plugin>
      <groupId>org.kie</groupId>
      <artifactId>kie-maven-plugin</artifactId>
      <version>${kie.version}</version>
      <extensions>>true</extensions>
    </plugin>
  </plugins>
</build>
```

- プロジェクトで永続サポートのあるプロセスエンジンまたはデシジョンエンジンを使用する場合には、Red Hat Business Automation BOM ファイルから **version.org.hibernate-4ee7** プロパティをコピーして、**pom.xml** ファイルの **dependencyManagement** セクションに以下の Hibernate の依存関係を宣言する必要があります。

### Hibernate の依存関係

```
<!-- hibernate dependencies -->
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-entitymanager</artifactId>
      <version>${version.org.hibernate-4ee7}</version>
    </dependency>

    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-core</artifactId>
      <version>${version.org.hibernate-4ee7}</version>
    </dependency>
  </dependencies>
</dependencyManagement>
```

-

## 第8章 キーストアを使用したパスワードセキュリティの確保

キーストアを使用して、Business Central と KIE Server の間の通信に使用するパスワードを暗号化できます。コントローラーと KIE Server のパスワードを暗号化する必要があります。Business Central と KIE Server を別のアプリケーションサーバーにデプロイする場合には、いずれのアプリケーションサーバーもキーストアを使用する必要があります。

Java Cryptography Extension KeyStore (JCEKS) は、対称鍵をサポートするので、キーストアには JCEKS を使用してください。



### 注記

KIE Server が JCEKS で設定されていない場合には、KIE Server のパスワードはシステムプロパティにプレーンテキスト形式で保存されます。

### 前提条件

- KIE Server が IBM WebSphere Application Server にインストールされている。
- 「[KIE Server のグループおよびロールの作成](#)」の記載通りに、**kie-server** ロールを割り当てた KIE Server ユーザーが作成されている。
- Java 8 以降がインストールされている。

### 手順

1. JCEKS キーストアを作成します。
2. プロンプトが表示されたら、作成した KIE Server ユーザーのパスワードを入力します。
3. 以下の表に記載のシステムプロパティを設定します。

表8.1 KIE Server JCEKS を読み込む時に使用するシステムプロパティ

システムプロパティ	プレースホルダー	説明
<b>kie.keystore.keyStoreURL</b>	<KEYSTORE_URL>	使用する JCEKS の URL。例: <b>file:///home/kie/keystores/keystore.jceks</b>
<b>kie.keystore.keyStorePwd</b>	<KEYSTORE_PWD>	JCEKS のパスワード
<b>kie.keystore.key.server.alias</b>	<KEY_SERVER_ALIAS>	パスワードの保存先となる REST サービスのキーのエイリアス
<b>kie.keystore.key.server.pwd</b>	<KEY_SERVER_PWD>	保存したパスワードを使用する REST サービスのエイリアスのパスワード
<b>kie.keystore.key.ctrl.alias</b>	<KEY_CONTROL_ALIAS>	パスワードの保存先のデフォルトの REST Process Automation Controller のキーのエイリアス

システムプロパティ	プレースホルダー	説明
<b>kie.keystore.key.ctrl.pwd</b>	<b>&lt;KEY_CONTROL_PWD&gt;</b>	保存したパスワードを使用する、デフォルトの REST Process Automation Controller のエイリアスのパスワード

4. KIE Server を起動して、設定を検証します。

## 第9章 次のステップ

- 『[デシジョンサービスのスタートガイド](#)』
- 『[ガイド付きデシジョンテーブルを使用したデシジョンサービスの作成](#)』

## 付録A バージョン情報

本ドキュメントの最終更新日: 2020 年 3 月 18 日 (水)