Red Hat Decision Manager 7.4

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

ガイド

Last Updated: 2023-03-08

Red Hat Decision Manager 7.4 Red Hat JBoss Web Server への Red Hat Decision Manager のインストールおよび設定

ガイド

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 © 2022 | You need to change the HOLDER entity in the en-US/Installing_and_configuring_Red_Hat_Decision_Manager_on_Red_Hat_JBoss_Web_Server.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 ® is the registered trademark of Linus Torvalds in the United States and other countries.

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

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

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

Node.js ® is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

本書では、Red Hat JBoss Web Server 5.1.0 に Red Hat Decision Manager 7.4 をインストールする方法を説明します。

目次

はじめに 3

第1章 RED HAT DECISION MANAGER について 4
1.1. RED HAT DECISION MANAGER のコンポーネント 4
1.2. RED HAT DECISION MANAGER ロールおよびユーザー 5

第2章 RED HAT DECISION MANAGER インストールファイルのダウンロード 7

第3章 RED HAT DECISION MANAGER インストーラーの使用方法 8
3.1. インタラクティブモードでのインストーラーの使用 8
3.2. CLI モードでのインストーラーの使用 10

第4章 DECISION SERVER ZIP ファイルのインストールと設定 12
4.1. ZIP ファイルからの DECISION SERVER のインストール 12

第5章 キーストアを使用したパスワードセキュリティーの確保 14

第6章 DECISION SERVER インストールの確認 16

第7章 ヘッドレス DECISION MANAGER コントローラーのダウンロードおよびインストール 17

第8章 スタンドアロン BUSINESS CENTRAL の実行 20
8.1. サポートされるプロパティー 21

第9章 RED HAT DECISION MANAGER の MAVEN 設定およびリポジトリー 26
9.1. プロジェクト設定ファイル (POM.XML) を使用した MAVEN の設定 26
9.2. MAVEN 設定ファイルの修正 26
9.3. RED HAT DECISION MANAGER の MAVEN 依存関係の追加 27
9.4. オフラインで使用する MAVEN ミラーリポジトリーの用意 29

第10章 GIT リポジトリーからのプロジェクトのインポート 32

第11章 LDAP と SSL の統合 33

付録A バージョン情報 34

はじめに

本書では、JBoss Web Server に Red Hat Decision Manager 7.4 をインストールする方法を説明します。

前提条件

- **Red Hat Decision Manager インストールの計画**の内容を確認している。
- Red Hat JBoss Web Server 5.1.0 がインストールされている。Red Hat JBoss Web Server のインストールに関する情報は、**Red Hat JBoss Web Server インストールガイド**を参照してください。

第1章 RED HAT DECISION MANAGER について

Red Hat Decision Manager は、ビジネスルール管理、複合イベント処理、Decision Model & Notation (DMN) 実行、およびプランニングの問題を解決するための Business Optimizer を組み合わせた、オープンソースの意思決定管理プラットフォームです。これにより、ビジネス上の意思決定を自動化し、そのロジックをビジネス全体で利用できるようにします。

ルール、デシジョンテーブル、DMN モデルなどのビジネスアセットはプロジェクト内で組織化され、Business Central リポジトリーに保存されます。これにより、ビジネス全体で一貫性や透明性を維持し、監査を行えます。ビジネスユーザーは、IT 担当者からのサポートなしでビジネスロジックを編集できます。

Red Hat JBoss Web Server は Tomcat をベースとしたエンタープライズレベルの Web サーバーで、中規模および大規模のアプリケーション用に設計されています。Red Hat JBoss Web Server は、Java Server Pages (JSP) および Java Servlet テクノロジー、PHP、ならびに CGI をデプロイするための単一プラットフォームを提供します。

Red Hat JBoss Web Server インストールに Decision Server およびヘッドレス Decision Manager コントローラーをインストールできます。または、スタンドアロンの Business Central JAR ファイルを実行できます。

本書では、Red Hat JBoss Web Server インスタンスに Red Hat Decision Manager をインストールする方法を説明します。

その他の環境に Red Hat Decision Manager をインストールする方法については、以下のドキュメントを参照してください。

- **Red Hat JBoss EAP 7.2 への Red Hat Decision Manager のインストールおよび設定**
- **IBM WebSphere Application Server への Decision Server のインストールおよび設定**
- **Oracle WebLogic Server への Decision Server のインストールおよび設定**
- **Red Hat OpenShift Container Platform への Red Hat Decision Manager イミュータブルサーバー環境のデプロイメント**
- **Red Hat OpenShift Container Platform への Red Hat Decision Manager オーサリングまたは管理サーバー環境のデプロイメント**
- **Operator を使用した Red Hat OpenShift Container Platform への Red Hat Decision Manager 環境のデプロイメント**

サポートされるコンポーネントについては、以下のドキュメントを参照してください。

- What is the mapping between Red Hat Decision Manager and the Maven library version?
- Red Hat Decision Manager 7 でサポートされる設定

1.1. RED HAT DECISION MANAGER のコンポーネント

Red Hat Decision Manager は、Decision Central と Decision Server で設定されます。

- Business Central は、ビジネスルールを作成して管理するグラフィカルユーザーインターフェイスです。Business Central は、Red Hat JBoss EAP インスタンスまたは Red Hat OpenShift Container Platform (OpenShift) にインストールできます。

Business Central は、スタンドアロンの JAR ファイルとしても使用できます。Business Central スタンドアロンの JAR ファイルとして使用して、アプリケーションサーバーにデプロイせずに Business Central を実行できます。

- Decision Server では、ルール、およびその他のアーティファクトが実行されます。これは、ルールをインスタンス化して実行し、計画の問題を解決するために使用されます。Decision Server は、Red Hat JBoss EAP インスタンス、OpenShift、Oracle WebLogic Server インスタンス、IBM WebSphere Application Server インスタンスに、または Spring Boot アプリケーションの一部としてインストールできます。
  Decision Server は、管理モードまたは非管理モードで動作するように設定できます。非管理モードの場合は、手動で KIE コンテナー (デプロイメントユニット) を作成および維持する必要があります。KIE コンテナーは、プロジェクトの特定のバージョンです。管理モードの場合は、Decision Manager コントローラーが Decision Server の設定を管理し、ユーザーはコントローラーと対話形式で KIE コンテナーを作成、維持します。

  Red Hat JBoss Web Server インストールに Decision Server およびヘッドレス Decision Manager コントローラーをインストールできます。または、スタンドアロンの Business Central JAR ファイルを実行できます。

1.2. RED HAT DECISION MANAGER ロールおよびユーザー

Business Central または Decision Server にアクセスするには、サーバーを起動する前にユーザーを作成して適切なロールを割り当てます。

Business Central と Decision Server は、JAVA 認証承認サービス (JAAS) ログインモジュールを使用してユーザーを認証します。Business Central と Decision Server の両方が単一のインスタンスで実行されている場合は、同じ JAAS サブジェクトとセキュリティードメインを共有します。したがって、Business Central に対して認証されたユーザーは、Decision Server にもアクセスできます。

ただし、Business Central と Decision Server が異なるインスタンスで実行されている場合、JAAS ログインモジュールは両方に対して個別にトリガーされます。したがって、Business Central で認証されたユーザーは、Decision Server にアクセス (Business Central でプロセス定義を表示または管理など) するための個別認証が必要となります。ユーザーが Decision Server で認証されていない場合は、ログファイルに 401 エラーが記録され、Business Central に **Invalid credentials to load data from remote server.Contact your system administrator.** メッセージが表示されます。

本セクションでは、利用可能な Red Hat Decision Manager のユーザーロールを説明します。

注記

admin、**analyst**、および **rest-all** のロールは Business Central 用に予約されています。**kie-server** ロールは Decision Server 用に予約されています。このため、Business Central または Decision Server のいずれか、またはそれら両方がインストールされているかどうかによって、利用可能なロールは異なります。

- **admin**: **admin** ロールを持つユーザーは Business Central 管理者です。管理者は、ユーザーの管理や、リポジトリーの作成、クローン作成、および管理ができます。アプリケーションで必要な変更をすべて利用できます。**admin** ロールを持つユーザーは、Red Hat Decision Manager の全領域にアクセスできます。

- **analyst**: **analyst** ロールを持つユーザーには、すべてのハイレベル機能へのアクセスがあります。プロジェクトのモデル化が可能です。ただし、このユーザーは、**Design → Projects** ビューでスペースに貢献者を追加したり、スペースを削除したりできません。**analyst** ロール

を持つユーザーは、管理者向けの **Deploy → Execution Servers** ビューにアクセスできません。ただし、これらのユーザーは、ライブラリーパースペクティブにアクセスするときに **Deploy** ボタンを使用できます。

- **rest-all**: **rest-all** ロールを持つユーザーは、Business Central REST 機能にアクセスできます。
- **kie-server**: **kie-server** ロールを持つユーザーは Decision Server (KIE サーバー) REST 機能へのアクセスがあります。このロールは、Business Central で Manage ビューおよび Track ビューにアクセスするユーザーに必要になります。

第2章 RED HAT DECISION MANAGER インストールファイルのダウンロード

お使いの環境およびインストール要件に応じた Red Hat Decision Manager ディストリビューションをダウンロードします。

手順

1. Red Hat カスタマーポータルの 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。
   - **Product:** Decision Manager
   - **バージョン:** 7.4
2. お好みのインストール方法に従って、以下の製品ディストリビューションのいずれかをダウンロードします。

   注記

   ダウンロードするのは、どれか 1 つのディストリビューションのみです。

   - インストーラーを使用して Red Hat JBoss Web Server に Red Hat Decision Manager をインストールする場合は、**Red Hat Decision Manager 7.4.0 Installer**(**rhdm-installer-7.4.0.jar**) をダウンロードします。インストーラーのグラフィックユーザーインターフェイスに従って、インストールプロセスを進めます。
   - デプロイ可能な ZIP ファイルを使用して Red Hat JBoss Web Server 上に Decision Server をインストールするには、**Red Hat Decision Manager 7.4.0 Add Ons**(**rhdm-7.4.0-add-ons.zip**) ファイルをダウンロードします。
     ZIP ファイルはグラフィカルユーザーインターフェイスを必要としないため、SSH を使用して Red Hat Decision Manager をインストールできます。
   - アプリケーションサーバーにデプロイせずに Business Central を実行するには、**Red Hat Decision Manager 7.4.0 Business Central Standalone** (**rhdm-7.4.0-decision-central-standalone.jar**) をダウンロードします。

第3章 RED HAT DECISION MANAGER インストーラーの使用方法

本セクションでは、インストール JAR ファイルを使用して Decision Server およびヘッドレス Decision Manager コントローラーをインストールする方法を説明します。JAR ファイルは、既存の Red Hat JBoss Web Server 5.1.0 サーバーインストール環境に Red Hat Decision Manager をインストールする実行ファイルです。インストーラーは、インタラクティブモードまたはコマンドラインインターフェイス (CLI) モードで実行できます。

次のステップ:

以下のセクションのいずれかに記載される手順を行います。

- 「インタラクティブモードでのインストーラーの使用」
- 「CLI モードでのインストーラーの使用」

3.1. インタラクティブモードでのインストーラーの使用

Red Hat Decision Manager のインストーラーは、実行可能な JAR ファイルです。このインストーラーを使用して、既存の Red Hat JBoss Web Server 5.1.0 のサーバーインストールに Red Hat Decision Manager をインストールできます。

注記

セキュリティー上の理由で、root 以外のユーザーでインストーラーを実行する必要があります。

前提条件

- バックアップを作成してある Red Hat JBoss Web Server 5.1.0 サーバーインストールが利用できる。
- 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付与されている。

  注記

  Tomcat への書き込み権限のあるユーザーとしてログインします。

- **$PATH** 環境変数に含まれている JAR バイナリー。Red Hat Enterprise Linux では、 **java-$JAVA_VERSION-openjdk-devel** パッケージに含まれています。

  注記

  Red Hat Decision Manager は、UTF-8 エンコーディングで機能するように設計されています。基礎となる JVM で別のエンコーディングシステムを使用すると、予期せぬエラーが発生する可能性があります。JVM で UTF-8 を使用するようにするには、**-Dfile.encoding=UTF-8** のシステムプロパティーを使用します。

手順

1. 端末ウインドウで、インストーラー JAR ファイルをダウンロードしたディレクトリーに移動し、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.4.0.jar
```

–

注記

Windows でインストーラーを実行すると、インストール時に管理者の認証情報が求められる場合があります。この要求を回避するには、インストールコマンドに **izpack.mode=privileged** オプションを追加します。

```
java -Dizpack.mode=privileged -jar
rhdm-installer-7.4.0.jar
```

また、32 ビットの Java 仮想マシンでインストーラーを実行している場合には、メモリー不足になる可能性があります。この問題を防ぐには、以下のコマンドを実行します。

```
java -XX:MaxHeapSize=4g -jar
rhdm-installer-7.4.0.jar
```

グラフィカルインストーラーにスプラッシュ画面と使用許諾契約書のページが表示されます。

2. **I accept the terms of this license agreement (本使用許諾契約書の内容に同意します)**をクリックし、**Next** をクリックします。

3. Red Hat Decision Manager をインストールする Red Hat JBoss Web Server 5.1.0 サーバーのホームを指定して、**Next** をクリックします。

4. インストールするコンポーネントを選択し、**Next** をクリックします。
Business Central は Red Hat JBoss Web Server にインストールすることができません。Business Central をインストールできるのは Red Hat JBoss EAP だけです。ヘッドレス Decision Manager コントローラーは、Decision Server を管理するために使用します。ヘッドレス Decision Manager コントローラーは、Decision Server の管理に使用されます。複数の Decision Server インスタンスを管理する予定がある場合は、ヘッドレス Decision Manager コントローラーをインストールします。

5. ユーザーを作成して **Next** をクリックします。デフォルトでは、新規ユーザーに **admin** ロール、**kie-server** ロール、および **rest-all** ロールが付与されます。Decision Server REST 機能にアクセスするには **kie-server** ロールが必要です。

注記

必ず、既存のユーザー、ロール、またはグループとは異なるユーザー名を指定してください。たとえば、**admin** という名前のユーザーは作成しないでください。

パスワードは 8 文字以上で、数字と、英数字以外の文字をそれぞれ 1 文字以上使用する必要があります。ただし & の文字は使用できません。

ユーザー名とパスワードを書き留めておきます。Business Central および Decision Server にアクセスする際に必要になります。

6. **Installation Overview** ページで **Next** をクリックしてインストールを起動します。Installation Overview ページに、インストールされるコンポーネントの一覧が表示されます。

7. インストールが完了したら、**Next** をクリックします。

8. 画面上部に **Processing finished** が表示されたら、**Next** をクリックしてインストールを完了します。

9. 必要であれば、**Generate Installation Script and Properties File**をクリックしてインストールデータを XML ファイルに保存し、続いて **Done** をクリックします。このファイルを使用して、同じタイプのサーバー上に Red Hat Decision Manager を自動的にインストールすることができます。XML ファイルの **installpath** パラメーターを変更して、Red Hat Decision Manager をインストールする新しいサーバーのパスを指定しなければならない点に注意してください。XML ファイルを使用してインストールを行う場合は、以下のコマンドを実行します。

```
java -jar rhdm-installer-7.4.0.jar <path-to-file>
```

インストーラーを使用して Red Hat Decision Manager が正常にインストールされました。

3.2. CLI モードでのインストーラーの使用

コマンドラインインターフェイス (CLI) を使用して Red Hat Decision Manager インストーラーを実行できます。

注記

セキュリティー上の理由で、root 以外のユーザーでインストーラーを実行する必要があります。

前提条件

- バックアップを作成してある Red Hat JBoss Web Server 5.1.0 サーバーインストールが利用できる。
- 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付与されている。

  注記

  Tomcat への書き込み権限のあるユーザーとしてログインします。

- **$PATH** 環境変数に含まれている JAR バイナリー。Red Hat Enterprise Linux では、 **java-$JAVA_VERSION-openjdk-devel** パッケージに含まれています。

  注記

  Red Hat Decision Manager は、UTF-8 エンコーディングで機能するように設計されています。基礎となる JVM で別のエンコーディングシステムを使用すると、予期せぬエラーが発生する可能性があります。JVM で UTF-8 を使用するようにするには、**-Dfile.encoding=UTF-8** のシステムプロパティーを使用します。

手順

1. 端末ウインドウにおいて、インストーラーファイルをダウンロードしたディレクトリーに移動し、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.4.0.jar -console
```

   コマンドラインの対話プロセスが開始し、使用許諾契約書が表示されます。

```
press 1 to continue, 2 to quit, 3 to redisplay.
```

2. 使用許諾契約書を読んで **1** を入力し、Enter キーを押して続行します。

```
Specify the home directory of one of the following servers:  Red Hat JBoss EAP 7.2 or Red
Hat JBoss Web Server 5.1.0
```

3. 既存の Red Hat JBoss Web Server 5.1.0 インストールの親ディレクトリーを入力します。インストーラーが、指定したインストール場所を確認します。**1** を入力して確認し、続行します。

4. インストーラーの指示に従って、インストールを完了します。

注記

ユーザー名およびパスワードを作成する場合は、指定したユーザー名が既存のロールまたはグループの名前と競合しないようにしてください。たとえば、**admin** というロールがある場合は、**admin** という名前のユーザーは作成しないでください。

パスワードは 8 文字以上で、数字と、英数字以外の文字をそれぞれ 1 文字以上使用する必要があります。ただし **&** の文字を使用することは **できません**。

ユーザー名とパスワードを書き留めておきます。Business Central および Decision Server にアクセスする際に必要になります。

5. インストールが完了すると、以下のメッセージが表示されます。

```
Would you like to generate an automatic installation script and properties file?
```

6. **y** を入力してインストールデータが含まれる XML ファイルを作成します。あるいは、**n** を入力してインストールを完了します。**y** を入力すると、XML ファイルのパスを指定するよう求められます。

7. パスを入力するか、Enter キーを押して提案されたパスを了承します。

8. Decision Server だけをインストールした場合は、この手順を繰り返して、別のサーバーにヘッドレス Decision Manager コントローラーをインストールします。

第4章 DECISION SERVER ZIP ファイルのインストールと設定

カスタマーポータルの **Red Hat Decision Manager 7.4.0 Add Ons**(**rhdm-7.4.0-add-ons.zip**) ファイルで利用可能な **rhdm-7.4-kie-server-jws.zip** ファイルを使用して Decision Server をインストールできます。

4.1. ZIP ファイルからの DECISION SERVER のインストール

Decision Server はビジネスアセットのランタイム環境を提供し、アセットレポジトリー (ナレッジストア) に保存されたデータにアクセスします。ZIP ファイルを使用して既存の Red Hat JBoss Web Server 5.1.0 サーバーインスタンスに Decision Server をインストールすることができます。

注記

インストーラー JAR ファイルを使用した Decision Server のインストール方法は *3章Red Hat Decision Manager インストーラーの使用方法* を参照してください。

- *2章Red Hat Decision Manager インストールファイルのダウンロード* に従って **Red Hat Decision Manager 7.4.0 Add Ons** (**rhdm-7.4.0-add-ons.zip**) ファイルがダウンロードされている。
- バックアップを作成してある Red Hat JBoss Web Server 5.1.0 サーバーインストールが利用できる。Red Hat JBoss Web Server インストールのベースディレクトリーは **JWS_HOME** とします。
- 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付与されている。

手順

1. **rhdm-7.4.0-add-ons.zip** ファイルを展開します。
2. 展開した **rhdm-7.4.0-add-ons.zip** ファイルから、以下のファイルを展開します。
   - **rhdm-7.4-kie-server-jws.zip**
   - **rhdm-7.4-decision-engine.zip**

   以下の説明では、展開した **rhdm-7.4-kie-server-jws.zip** ファイルが含まれるディレクトリーは **JWS_TEMP_DIR** と呼ばれ、展開した **rhdm-7.4-decision-engine.zip** ファイルが含まれるディレクトリーは **ENGINE_TEMP_DIR** と呼ばれます。
3. **JWS_TEMP_DIR/rhdm-7.4-kie-server-jws/kie-server.war** ディレクトリーを **JWS_HOME/tomcat/webapps** ディレクトリーにコピーします。

   注記

   コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss Web Server インスタンスの既存デプロイメントと競合しないことを確認します。
4. **kie-server.war** ディレクトリーから **.war** 拡張子を削除します。
5. **kie-tomcat-integration-7.23.0.Final-redhat-00002.jar** ファイルを **ENGINE_TEMP_DIR** ディレクトリーから **JWS_HOME/tomcat/lib** ディレクトリーに移動します。

6. **jboss-jacc-api-<VERSION>.jar** ファイル、**slf4j-api-<VERSION>.jar** ファイル、および **slf4j-jdk14-<VERSION>.jar** ファイルを **ENGINE_TEMP_DIR/lib** ディレクトリーから **JWS_HOME/tomcat/lib** ディレクトリーに移動します。**<VERSION>** は **lib** ディレクトリーのバージョンのアーティファクトファイル名に置き換えます。

7. 最後の Valve 定義の後にある **JWS_HOME/conf/server.xml** ファイルの **<host>** 要素に、以下の行を追加します。

```
<Valve className="org.kie.integration.tomcat.JACCValve" />
```

8. テキストエディターで **JWS_HOME/tomcat/conf/tomcat-users.xml** ファイルを開きます。

9. **JWS_HOME/tomcat/conf/tomcat-users.xml** ファイルにユーザーおよびロールを追加します。以下の例で、**<ROLE_NAME>** は Red Hat Decision Manager でサポートされるロールです。**<USER_NAME>** および **<USER_PWD>** は、選択したユーザー名とパスワードです。

```
<role rolename="<ROLE_NAME>"/>
<user username="<USER_NAME>" password="<USER_PWD>" roles="<ROLE_NAME>"/>
```

ユーザーに複数のロールを割り当てる場合は、以下の例に示すようにロールをコンマで区切ります。

```
<role rolename="admin"/>
<role rolename="kie-server"/>
<user username="rhdmUser" password="user1234" roles="admin,kie-server"/>
```

10. **JWS_HOME/tomcat/bin** ディレクトリーで以下の手順の 1 つを実行します。

- Linux または UNIX の場合は、以下の内容の **setenv.sh** ファイルを作成します。

```
CATALINA_OPTS="-Xmx1024m
 -Dorg.jboss.logging.provider=jdk
 -Dorg.jbpm.server.ext.disabled=true
 -Dorg.jbpm.ui.server.ext.disabled=true
 -Dorg.jbpm.case.server.ext.disabled=true"
```

- Windows の場合は、以下の内容の **setenv.bat** ファイルを作成します。

```
set CATALINA_OPTS="-Xmx1024m -Dorg.jboss.logging.provider=jdk -
Dorg.jbpm.server.ext.disabled=true -Dorg.jbpm.ui.server.ext.disabled=true -
Dorg.jbpm.case.server.ext.disabled=true
```

第5章 キーストアを使用したパスワードセキュリティーの確保

キーストアを使用して、Business Central と Decision 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 Decision Server のパスワードを暗号化する必要があります。Business Central と Decision 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ため、キーストアには JCEKS を使用してください。JDK インストールに含まれる KeyTool を使用して、新しい JCEKS を作成します。

注記

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

前提条件

- Red Hat JBoss Web Server に Decision Server がインストールされている。
- Java 8 以降がインストールされている。

手順

1. テキストエディターで **JWS_HOME/tomcat/conf/tomcat-users.xml** ファイルを開きます。
2. **kie-server** ロールを割り当てた Decision Server ユーザーを **JWS_HOME/tomcat/conf/tomcat-users.xml** ファイルに追加します。以下の例では、**<USER_NAME>** と **<PASSWORD>** は任意のユーザー名とパスワードに置き換えます。

```
<role rolename="kie-server"/>
<user username="<USER_NAME>" password="<PASSWORD>" roles="kie-server"/>
```

3. KeyTool を使用して JCEKS を作成するには、Java 8 のホームディレクトリーで以下のコマンドを実行します。

```
$<JAVA_HOME>/bin/keytool -importpassword -keystore <KEYSTORE_PATH> -keypass
<ALIAS_KEY_PASSWORD> -alias <PASSWORD_ALIAS> -storepass
<KEYSTORE_PASSWORD> -storetype JCEKS
```

   上記の例では、以下の変数を置き換えてください。

   - **<KEYSTORE_PATH>**: キーストアの保存先のパス
   - **<KEYSTORE_PASSWORD>**: キーストアのパスワード
   - **<ALIAS_KEY_PASSWORD>**: エイリアスで保存した値にアクセスする時に使用するパスワード
   - **<PASSWORD_ALIAS>**: プロセスへのエントリーに使用するエイリアス

4. プロンプトが表示されたら、作成した Decision Server ユーザーのパスワードを入力します。
5. システムプロパティーを設定するには、**JWS_HOME/tomcat/bin** ディレクトリーで以下の手順の 1 つを実行し、変数は以下の表で記載されているように置き換えます。

- Linux または UNIX の場合は、以下の内容の **setenv.sh** ファイルを作成します。

```
set CATALINA_OPTS="
-Dkie.keystore.keyStoreURL=<KEYSTORE_URL>
-Dkie.keystore.keyStorePwd=<KEYSTORE_PWD>
-Dkie.keystore.key.server.alias=<KEY_SERVER_ALIAS>
-Dkie.keystore.key.server.pwd=<KEY_SERVER_PWD>
-Dkie.keystore.key.ctrl.alias=<KEY_CONTROL_ALIAS>
-Dkie.keystore.key.ctrl.key.ctrl.pwd=<KEY_CONTROL_PWD>
```

- Windows の場合は、以下の内容の **setenv.bat** ファイルを作成します。

```
set CATALINA_OPTS="
-Dkie.keystore.keyStoreURL=<KEYSTORE_URL>
-Dkie.keystore.keyStorePwd=<KEYSTORE_PWD>
-Dkie.keystore.key.server.alias=<KEY_SERVER_ALIAS>
-Dkie.keystore.key.server.pwd=<KEY_SERVER_PWD>
-Dkie.keystore.key.ctrl.alias=<KEY_CONTROL_ALIAS>
-Dkie.keystore.key.ctrl.key.ctrl.pwd=<KEY_CONTROL_PWD>
```

表5.1 Decision Server JCEKS を読み込む時に使用するシステムプロパティー

システムプロパティー	プレースホルダー	説明
kie.keystore.keyStoreURL	**<KEYSTORE_URL>**	使用する JCEKS の URL (例: **file:///home/kie/keystores/keystore.jceks**)
kie.keystore.keyStorePwd	**<KEYSTORE_PWD>**	JCEKS のパスワード
kie.keystore.key.server.alias	**<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
kie.keystore.key.server.pwd	**<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
kie.keystore.key.ctrl.alias	**<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
kie.keystore.key.ctrl.key.ctrl.pwd	**<KEY_CONTROL_PWD>**	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

6. Decision Server を起動して、設定を検証します。

第6章 DECISION SERVER インストールの確認

Decision Server が正しくインストールされていることを確認します。

前提条件

- Decision Server がインストールされ、設定されています。

手順

1. Decision Server を起動するには、**JWS_HOME/tomcat/bin** ディレクトリーからターミナルを開いて以下のコマンドのいずれかを入力します。
   - Linux または UNIX ベースのシステムの場合:

     ```
     $ ./startup.sh
     ```

   - Windows の場合:

     ```
     startup.bat
     ```

2. 数分後に **JWS_HOME/tomcat/logs** ディレクトリーのファイルを確認し、エラーが発生している場合は修正します。
3. Web ブラウザーに **http://localhost:8080/kie-server/services/rest/server** を入力し、Decision Server が正常に動作していることを確認します。
4. **tomcat-users.xml** ファイルに保存したユーザー名とパスワードを入力します。

第7章 ヘッドレス DECISION MANAGER コントローラーのダウンロードおよびインストール

Decision Server は、管理モードまたは非管理モードで動作するように設定できます。非管理モードの場合は、手動で KIE コンテナー (デプロイメントユニット) を作成および維持する必要があります。管理モードの場合は、Decision Manager コントローラーが Decision Server の設定を管理し、ユーザーはコントローラーと対話形式で KIE コンテナーを作成、維持します。

Decision Manager コントローラーは Business Central と統合します。Business Central をインストールしている場合は、**Execution Server** ページを使用して KIE コンテナーを作成および維持します。ただし、Business Central をインストールしない場合は、ヘッドレス Decision Manager コントローラーをインストールし、REST API または Decision Server Java Client API を使用してそのコントローラーと対話します。

前提条件

- *2章Red Hat Decision Manager インストールファイルのダウンロード* に従って **Red Hat Decision Manager 7.4.0 Add Ons** (**rhdm-7.4.0-add-ons.zip**) ファイルがダウンロードされている。
- Red Hat JBoss Web Server 5.1.0 サーバーインストールが利用できる。Red Hat JBoss Web Server インストールのベースディレクトリーは **JWS_HOME** とします。
- 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付与されている。

手順

1. **rhdm-7.4.0-add-ons.zip** ファイルを展開します。**rhdm-7.4-controller-jws.zip** ファイルは展開したディレクトリーにあります。
2. **rhdm-7.4-controller-jws.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、この名前を **TEMP_DIR** とします。
3. **TEMP_DIR/rhdm-7.4-controller-jws.zip/controller.war** ディレクトリーを **JWS_HOME/tomcat/webapps** ディレクトリーにコピーします。

   注記

   コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss Web Server インスタンスの既存デプロイメントと競合しないことを確認します。

4. **controller.war** ディレクトリーから **.war** 拡張子を削除します。
5. **TEMP_DIR/rhdm-7.4-controller-jws/SecurityPolicy/** ディレクトリーのコンテンツを **JWS_HOME/bin** にコピーします。
6. ファイルの上書きを確認するメッセージが表示されたら、**Yes** を選択します。
7. **JWS_HOME/tomcat/conf/tomcat-users.xml** ファイルに **kie-server** ロールおよびユーザーを追加します。以下の例で、**<USER_NAME>** と **<PASSWORD>** は、任意のユーザー名とパスワードです。

```
<role rolename="kie-server"/>
<user username="<USER_NAME>" password="<PASSWORD>" roles="kie-server"/>
```

8. Decision Server を実行するインスタンスの **JWS_HOME/tomcat/bin** ディレクトリーで、以下のタスクの 1 つを実行します。

   - Linux または UNIX の場合は、以下の内容の **setenv.sh** ファイルを作成します。

     ```
     CATALINA_OPTS="-Xmx1024m
      -Dorg.jbpm.server.ext.disabled=true
      -Dorg.jbpm.ui.server.ext.disabled=true
      -Dorg.jbpm.case.server.ext.disabled=true
      -Dorg.kie.server.controller.user=<CONTROLLER_USER>
      -Dorg.kie.server.controller.pwd=<CONTROLLER_PWD>
      -Dorg.kie.server.id=<KIE_SERVER_ID>
      -Dorg.kie.server.location=http://<HOST>:<PORT>/kie-server/services/rest/server
      -Dorg.kie.server.controller=http://<HOST>:<PORT>/controller/rest/controller"
     ```

   - Windows の場合は、以下の内容の **setenv.bat** ファイルを作成します。

     ```
     set CATALINA_OPTS= -Xmx1024m
      -Dorg.jbpm.server.ext.disabled=true
      -Dorg.jbpm.ui.server.ext.disabled=true
      -Dorg.jbpm.case.server.ext.disabled=true
      -Dorg.kie.server.controller.user=<CONTROLLER_USER>
      -Dorg.kie.server.controller.pwd=<CONTROLLER_PWD>
      -Dorg.kie.server.id=<KIE_SERVER_ID>
      -Dorg.kie.server.location=http://<HOST>:<PORT>/kie-server/services/rest/server
      -Dorg.kie.server.controller=http://<CONTROLLER_HOST>:
     <CONTROLLER_PORT>/controller/rest/controller
     ```

     上記の例では、以下の変数に置き換えてください。

   - **<CONTROLLER_USER>** および **<CONTROLLER_PWD>** を、この手順の最初の方で定義した **kie-server** ロールを持つユーザー名とパスワードに置き換えます。

   - **<KIE_SERVER_ID>** を一意の識別子に置き換えます。

   - **<CONTROLLER_HOST>:<CONTROLLER_PORT>** をコントローラーの IP アドレス (ホストとポート) に置き換えます。Decision Server とコントローラーで同じサーバーを使用する場合は、**<CONTROLLER_HOST>:<CONTROLLER_PORT>** は **localhost:8080** になります。

9. ヘッドレス Decision Manager コントローラーを実行しているインスタンスの **JWS_HOME/tomcat/bin** ディレクトリーに、以下の内容で読み取り可能な **setenv.sh** ファイルを作成します。ここで、**<USERNAME>** は Decision Server ユーザーで、**<USER_PWD>** はそのユーザーのパスワードになります。
   CATALINA_OPTS="-Dorg.kie.server.user=<USERNAME> -Dorg.kie.server.pwd=<USER_PWD>"

10. ヘッドレス Decision Manager コントローラーを起動するには、**JWS_HOME/tomcat/bin** ディレクトリーに以下のいずれかのコマンドを実行します。

    - Linux または UNIX ベースのシステムの場合:

```
$ ./startup.sh
```

- Windows の場合:

```
startup.bat
```

11. 数分後に **JWS_HOME/tomcat/logs** ディレクトリーを確認し、エラーが発生している場合には修正します。

12. ヘッドレス Decision Manager コントローラーが正常に動作していることを確認するには、Web ブラウザーに **http://<CONTROLLER_HOST>:<CONTROLLER_PORT>/controller/rest/controller/management/servers** を入力します。Decision Server とコントローラーで同じサーバーを使用する場合は、**<CONTROLLER_HOST>:<CONTROLLER_PORT>** は **localhost:8080** になります。

13. **tomcat-users.xml** ファイルに保存したユーザー名とパスワードを入力します。

第8章 スタンドアロン BUSINESS CENTRAL の実行

Business Central スタンドアロン JAR ファイルを使用して、Red Hat JBoss EAP などのアプリケーションサーバーにデプロイせずに Business Central を実行できます。

注記

Red Hat は、Red Hat Enterprise Linux にインストールした場合に限りこのインストールタイプをサポートします。

前提条件

- *2章Red Hat Decision Manager インストールファイルのダウンロード* の説明どおりに、**Red Hat Decision Manager 7.4.0 Business Central Standalone** (**rhdm-7.4.0-decision-central-standalone.jar**) ファイルがダウンロードされている。

手順

1. ディレクトリーを作成し、**rhdm-7.4.0-decision-central-standalone.jar** ファイルをこのディレクトリーに移動します。
2. ターミナルウィンドウで、スタンドアロン JAR ファイルを含むディレクトリーに移動します。
3. **application-users.properties** ファイルを作成し、管理者ユーザーを含めます。この Business Central インスタンスが Decision Server の Decision Manager コントローラーになる場合は、Decision Manager コントローラーユーザーを含めます。以下に例を示します。

```
rhdmAdmin=password1
controllerUser=controllerUser1234
```

4. **application-roles.properties** ファイルを作成し、**application-users.properties** ファイルに含まれるユーザーにロールを割り当てます。以下に例を示します。

```
rhdmAdmin=admin
controllerUser=kie-server
```

5. 以下の内容で **application-config.yaml** 設定ファイルを作成します。**<APPLICATION_USERS>** は **application-users.properties** ファイルへのパスで、**<APPLICATION_ROLES>** は **application-roles.properties** ファイルへのパスです。

```
thorntail:
  management:
    security-realms:
      ApplicationRealm:
        local-authentication:
          default-user: local
          allowed-users: local
          skip-group-loading: true
        properties-authentication:
          path: <APPLICATION_USERS>
          plain-text: true
        properties-authorization:
          path: <APPLICATION_ROLES>
datasource:
```

```
management:
  wildfly:
    admin: admin
```

6. 以下のコマンドを入力します。

```
java -jar rhdm-7.4.0-decision-central-standalone.jar -s application-config.yaml
```

また、このコマンドに **-D<property>=<value>** パラメーターを追加して、Business Central でサポートされる任意のプロパティーを設定することもできます。以下は例になります。

```
java -jar rhdm-7.4.0-decision-central-standalone.jar -s application-config.yaml -D<property>=
<value> -D<property>=<value>
```

詳細は、「サポートされるプロパティー」 を参照してください。

8.1. サポートされるプロパティー

このセクションに記載の Business Central のシステムプロパティーは **standalone*.xml** ファイルに渡されます。

または、スタンドアロン Business Central のインストール時に、以下のコマンドで、本セクションに列記するプロパティーを使用することができます。

```
java -jar rhdm-7.4.0-decision-central-standalone.jar -s application-config.yaml -D<property>=<value>
-D<property>=<value>
```

このコマンドで、**<property>** が以下に列記するプロパティーのいずれかで、**<value>** がそのプロパティーに割り当てる値です。

Git ディレクトリー

以下のプロパティーを使用して、Business Central Git ディレクトリーの場所と名前を設定します。

- **org.uberfire.nio.git.dir**: Decision Server Git ディレクトリーの場所。
- **org.uberfire.nio.git.dirname**: Decision Server Git ディレクトリーの名前。デフォルト値は **.niogit** です。

HTTP 経由の Git

次のプロパティーを使用して、HTTP 経由で Git リポジトリーにアクセスできるように設定します。

- **org.uberfire.nio.git.proxy.ssh.over.http**: SSH が HTTP プロキシーを使用するかどうかを指定します。デフォルト: **false**
- **http.proxyHost**: HTTP プロキシー-のホスト名を定義します。デフォルト: **null**
- **http.proxyPort**: HTTP プロキシーのホストポート (整数値) を定義します。デフォルト: **null**
- **org.uberfire.nio.git.http.enabled**: HTTP デーモンを有効または無効にします。デフォルト: **true**
- **org.uberfire.nio.git.http.host**: このデーモンは、HTTP デーモンが有効な場合にホストの識別子としてこのプロパティーを使用します。これは、HTTP 経由で Git リポジトリーにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサーブレットコ

ンテナーに依存します。デフォルト: **localhost**

- **org.uberfire.nio.git.http.hostname**: HTTP デーモンが有効な場合に、このデーモンはホスト名の識別子としてこのプロパティーを使用します。これは、HTTP 経由で Git リポジトリーにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサーブレットコンテナーに依存します。デフォルト: **localhost**

- **org.uberfire.nio.git.http.port**: このデーモンは、HTTP デーモンが有効な場合にポート番号としてこのプロパティーを使用します。これは、HTTP 経由で Git リポジトリーにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサーブレットコンテナーに依存します。デフォルト値: **8080**

HTTPS 経由の Git

次のプロパティーを使用して、HTTPS 経由で Git リポジトリーにアクセスできるように設定します。

- **org.uberfire.nio.git.proxy.ssh.over.https**: SSH が HTTPS プロキシーを使用するかどうかを指定します。デフォルト: **false**

- **https.proxyHost**: HTTPS プロキシーのホスト名。デフォルト: **null**

- **https.proxyPort**: HTTPS プロキシーのホストポート (整数値)。デフォルト: **null**

- **org.uberfire.nio.git.https.enabled**: HTTPS デーモンを有効または無効にします。デフォルト: **false**

- **org.uberfire.nio.git.https.host**: このデーモンは、HTTPS デーモンが有効な場合にホストの識別子としてこのプロパティーを使用します。これは、HTTPS 経由で Git リポジトリーにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサーブレットコンテナーに依存します。デフォルト: **localhost**

- **org.uberfire.nio.git.https.hostname**: このデーモンは、HTTPS デーモンが有効な場合にホスト名の識別子としてこのプロパティーを使用します。これは、HTTPS 経由で Git リポジトリーにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサーブレットコンテナーに依存します。デフォルト: **localhost**

- **org.uberfire.nio.git.https.port**: このデーモンは、HTTPS デーモンが有効な場合にポート番号としてこのプロパティーを使用します。これは、HTTPS 経由で Git リポジトリーにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサーブレットコンテナーに依存します。デフォルト値: **8080**

Git デーモン

次のプロパティーを使用して、Git デーモンを有効にして設定します。

- **org.uberfire.nio.git.daemon.enabled**: Git デーモンを有効または無効にします。デフォルト値は **true** です。

- **org.uberfire.nio.git.daemon.host**: Git デーモンが有効な場合は、このプロパティーをローカルホストの識別子として使用します。デフォルト値は **localhost** です。

- **org.uberfire.nio.git.daemon.hostname**: Git デーモンが有効な場合は、このプロパティーをローカルホスト名の識別子として使用します。デフォルト: **localhost**

- **org.uberfire.nio.git.daemon.port**: Git デーモンが有効な場合は、このプロパティーをポート番号として使用します。デフォルト値は **9418** です。

- **org.uberfire.nio.git.http.sslVerify**: Git リポジトリーを確認する SSL 証明書を有効または無効にします。デフォルト: **true**

注記

デフォルトポートまたは割り当てられたポートが既に使用されている場合は、別のポートが自動的に選択されます。ポートが利用可能であることを確認し、詳細についてはログをチェックします。

Git SSH

次のプロパティーを使用して、Git SSH デーモンを有効にして設定します。

- **org.uberfire.nio.git.ssh.enabled**: SSH デーモンを有効または無効にします。デフォルト値は **true** です。
- **org.uberfire.nio.git.ssh.host**: SSH デーモンが有効な場合は、このプロパティーをローカルホスト識別子として使用します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.ssh.hostname**: SSH デーモンが有効な場合は、このプロパティーをローカルホスト名の識別子として使用します。デフォルト: **localhost**
- **org.uberfire.nio.git.SSH.port**: SSH デーモンが有効な場合は、このプロパティーをポート番号として使用します。デフォルト値は **8001** です。

注記

デフォルトポートまたは割り当てられたポートが既に使用されている場合は、別のポートが自動的に選択されます。ポートが利用可能であることを確認し、詳細についてはログをチェックします。

- **org.uberfire.nio.git.ssh.cert.dir**: ローカルの証明書が保存される **.security** ディレクトリーの場所。デフォルト: 作業ディレクトリー。
- **org.uberfire.nio.git.ssh.passphrase**: SCP スタイルの URL を持つ Git リポジトリーのクローンを作成する場合に、オペレーティングシステムの公開キーストアにアクセスするためのパスフレーズ。たとえば、**git@github.com:user/repository.git** です。
- **org.uberfire.nio.git.ssh.algorithm**: SSH で使用されるアルゴリズム。デフォルト値は **RSA** です。
- **org.uberfire.nio.git.ssh.ciphers**: コンマ区切りの暗号化の文字列。利用可能な暗号化は **aes128-ctr**、**aes192-ctr**、**aes256-ctr**、**arcfour128**、**arcfour256**、**aes192-cbc**、**aes256-cbc** です。このプロパティーを使用しない場合は、すべての暗号化が読み込まれます。
- **org.uberfire.nio.git.ssh.macs**: コンマ区切りのメッセージ認証コード (MAC) の文字列。利用可能な MAC は **hmac-md5**、**hmac-md5-96**、**hmac-sha1**、**hmac-sha1-96**、**hmac-sha2-256**、**hmac-sha2-512** です。このプロパティーを使用しない場合は、すべての MAC が読み込まれます。

注記

RSA、または DSA 以外のアルゴリズムを使う場合は、Bouncy Castle JCE ライブラリーを使用するようにアプリケーションサーバーを設定します。

Decision Server ノードおよび Decision Manager コントローラー

以下のプロパティーを使用して Decision Manager コントローラーと Decision Server ノードの接続を設定します。

- **org.kie.server.controller**: Decision Manager コントローラーとの接続に使用される URL (例: **ws://localhost:8080/decision-central/websocket/controller**)
- **org.kie.server.user**: Decision Manager コントローラーから Decision Server ノードに接続する際に使用するユーザー名。このプロパティーが必要なのは、この Business Central システムを Decision Manager コントローラーとして使用する場合だけです。
- **org.kie.server.pwd**: Decision Manager コントローラーから Decision Server ノードに接続する時に使用するパスワード。このプロパティーが必要なのは、この Business Central システムを Decision Manager コントローラーとして使用する場合だけです。

Maven など

以下のプロパティーを使用して、Maven などの機能を設定します。

- **org.uberfire.gzip.enable**: **GzipFilter** 圧縮フィルターで Gzip の圧縮を有効にするか、または無効にします。デフォルト値: **true**
- **org.uberfire.metadata.index.dir**: Lucene の **.index** ディレクトリーが保存される場所。デフォルト: 作業ディレクトリー
- **org.uberfire.ldap.regex.role_mapper**: LDAP プリンシパル名をアプリケーションのロール名にマッピングするのに使用する regex パターン。プリンシパルの値がロール名に一致する場合はアプリケーションのロール名に置き換えられるため、変数ロールはパターンの一部でなければならない点に注意してください。デフォルト: 不使用
- **org.uberfire.sys.repo.monitor.disabled**: 設定モニターを無効にします。無効にした場合の影響を正しく理解していない場合は、無効にしないでください。デフォルト値は **false** です。
- **org.uberfire.secure.key**: パスワードの暗号化で使用するパスワード。デフォルト値: **org.uberfire.admin**
- **org.uberfire.secure.alg**: パスワードの暗号化で使用する暗号化アルゴリズム。デフォルト値: **PBEWithMD5AndDES**
- **org.uberfire.domain**: uberfire が使用するセキュリティードメイン名。デフォルト値: **ApplicationRealm**
- **org.guvnor.m2repo.dir**: Maven リポジトリーディレクトリーが保存される場所。デフォルト値: **<working-directory>/repositories/kie**
- **org.guvnor.project.gav.check.disabled**: グループ ID、アーティファクト ID、およびバージョン (GAV) のチェックを無効にします。デフォルト値は **false** です。
- **org.kie.build.disable-project-explorer**: Project Explorer で選択したプロジェクトの自動ビルドを無効にします。デフォルト値は **false** です。
- **org.kie.verification.disable-dtable-realtime-verification**: デシジョンテーブルのリアルタイム確認および検証を無効にします。デフォルト値は **false** です。
- **kie.maven.offline.force**: Maven のオフライン動作を強制します。true に設定すると、オンラインの依存関係解決が無効になります。デフォルト: **false**。

注記

このプロパティーは、Business Central にのみ使用してください。他のコンポーネントとランタイム環境を共有する場合は、設定を分離して、Business Central にだけ適用してください。

- **designerdataobjects**: データオブジェクトの機能を無効にします。このパラメーターの値を **false** に設定します。

第9章 RED HAT DECISION MANAGER の MAVEN 設定およびリポジトリー

Red Hat Decision Manager プロジェクトの作成時に、Business Central は Business Central 用に設定された Maven リポジトリーを使用します。Maven project object model (POM) ファイル (**pom.xml**) を変更することで、Maven グローバルまたはユーザー設定を使用し、すべての Red Hat Decision Manager プロジェクトに対して Red Hat Decision Manager の公開リポジトリーから依存関係を取得するように指示できます。また、Business Central と Decision Server が外部の Maven リポジトリーを使用するか、Maven のミラーを準備してオフラインで使用できるように設定できます。

Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイメントオプションに関する詳細は、**Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイ** を参照してください。

9.1. プロジェクト設定ファイル (POM.XML) を使用した MAVEN の設定

Maven を使用して Red Hat Decision Manager プロジェクトをビルドおよび管理するには、POM ファイル (**pom.xml**) を作成および設定する必要があります。このファイルにはプロジェクトの設定情報が含まれます。詳細は Apache Maven Project を参照してください。

手順

1. Maven プロジェクトを生成します。**pom.xml** ファイルは、Maven プロジェクトの作成時に自動的に生成されます。

2. **pom.xml** ファイルを編集して、追加の依存関係および新たなリポジトリーを追加します。プロジェクトのコンパイルおよびパッケージング時に、Maven がすべての JAR ファイルおよび依存関係にある JAR ファイル を Maven リポジトリーからダウンロードします。

pom.xml ファイルのスキーマは、http://maven.apache.org/maven-v4_0_0.xsd から取得できます。POM ファイルの詳細は Apache Maven Project POM を参照してください。

9.2. MAVEN 設定ファイルの修正

Red Hat Decision Manager では、Maven の **settings.xml** ファイルを使用して Maven の実行を設定します。**settings.xml** ファイルにプロファイルを作成およびアクティベートし、Red Hat Decision Manager プロジェクトが使用する Maven リポジトリーを宣言します。

Maven の **settings.xml** ファイルの詳細は Apache Maven Project の Setting Reference を参照してください。

手順

1. **settings.xml** ファイルでは、Red Hat Decision Manager プロジェクトが使用するリポジトリーを宣言します。通常、これは Red Hat カスタマーポータルからダウンロードする オンラインの Red Hat Decision Manager Maven リポジトリー または Red Hat Decision Manager Maven リポジトリーと、使用するカスタムアーティファクトのリポジトリーです。

2. Business Central または Decision Server が **settings.xml** ファイルを使用するように設定されていることを確認します。たとえば、**kie.maven.settings.custom=<SETTINGS_FILE_PATH>** プロパティーを指定し、**<SETTINGS_FILE_PATH>** は **settings.xml** ファイルのパスに置き換えます。
   Red Hat JBoss Web Server で、Decision Server の場合は **-Dkie.maven.settings.custom=<SETTINGS_FILE_PATH>** を **setenv.sh** (Linux) または **setenv.bat** (Windows) ファイルの **CATALINA_OPTS** セクションに追加します。スタンドアロンの Business Central の場合は、以

下のコマンドを入力します。

```
java -jar rhpam-7.4.0-business-central-standalone.jar -s application-config.yaml -
Dkie.maven.settings.custom=<SETTINGS_FILE_PATH>
```

9.3. RED HAT DECISION MANAGER の MAVEN 依存関係の追加

ご自分の Red Hat Decision Manager プロジェクトで適切な Maven 依存関係を使用するには、プロジェクトの **pom.xml** ファイルに Red Hat Business Automation の BOM (bill of materials) ファイルを追加します。Red Hat Business Automation BOM は、Red Hat Decision Manager と Red Hat Process Automation Manager の両方に適用されます。BOM ファイルを追加すると、提供される Maven リポジトリーから、推移的依存関係の適切なバージョンがプロジェクトに含められます。

Red Hat Business Automation BOM (Bill of Materials) の詳細情報は、What is the mapping between Red Hat Decision 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>7.4.0.GA-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。

3. Business Central に作成されない標準のプロジェクトでは、お使いのプロジェクトに必要な依存関係をすべて指定します。Business Central に作成するプロジェクトでは、基本的なデシジョンエンジンの依存関係が Business Central に自動的に提供されます。

   - 標準的な Red Hat Decision 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。

   - 標準的な Red Hat Decision Manager プロジェクトでは、以下の依存関係を宣言します。

     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>
```

- Decision 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 Decision Server 依存関係

```
<dependency>
  <groupId>org.kie.server</groupId>
  <artifactId>kie-server-client</artifactId>
</dependency>
```

- Red Hat Decision 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

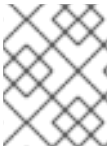
```
<dependency>
  <groupId>org.uberfire</groupId>
  <artifactId>uberfire-rest-client</artifactId>
</dependency>
```

- 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** と指定し、**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 Decision 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>
```

9.4. オフラインで使用する MAVEN ミラーリポジトリーの用意

公開インターネットへの送信アクセスが、Red Hat Decision Manager のデプロイメントに設定されていない場合は、必要なすべてのアーティファクトのミラーが含まれる Maven リポジトリーを用意して、このリポジトリーを使用できるようにする必要があります。



注記

Red Hat Decision Manager デプロイメントがインターネットに接続されている場合は、この手順を実行する必要はありません。

前提条件

- 公開インターネットへの送信アクセスが設定されているコンピューターが利用できる。

手順

1. 公開インターネットに送信アクセスができるコンピューターで、以下のアクションを実行します。

   a. 最新版の Offliner tool をダウンロードします。

   b. Red Hat カスタマーポータルの Software Downloads ページから利用可能な **rhdm-7.4.0-offliner.txt** の製品配信可能ファイルをダウンロードします。

   c. 以下のコマンドを入力して、Offliner ツールを使用し、必要なアーティファクトをダウンロードします。

   ```
   java -jar offliner-<version>.jar -r https://maven.repository.redhat.com/ga/ -r
   https://repo1.maven.org/maven2/ -d /home/user/temp rhdm-7.4.0-offliner.txt
   ```

   /home/user/temp は空の一時ディレクトリーに、**<version>** はダウンロードした Offliner ツールのバージョンに置き換えます。ダウンロードにはかなり時間がかかる可能性があります。

   d. このツールにより、ダウンロードの失敗が報告された場合には、以下のコマンドを入力して初回にダウンロードに失敗したアーティファクトをダウンロードします。

   ```
   grep Path: errors.log | sed -n -e 's/^.*Path: //p' > failed-downloads.txt
   java -jar offliner-<version>.jar -r https://maven.repository.redhat.com/ga/ -r
   https://repo1.maven.org/maven2/ -d /home/user/temp failed-downloads.txt
   ```

もう一度ダウンロードの失敗が報告され、失敗数が初回のダウンロード合計数と比較して少しの場合には、続行してください。

2. Business Central 外でサービスを開発し、追加の依存関係がある場合は、ミラーリポジトリーにその依存関係を追加します。サービスを Maven プロジェクトとして開発した場合は、以下の手順を使用し、これらの依存関係を自動的に用意します。公開インターネットへに送信接続できるコンピューターで、この手順を実行します。

   a. ローカルの Maven キャッシュディレクトリー (~/**.m2/repository**) のバックアップを作成して、ディレクトリーを削除します。

   b. **mvn clean install** コマンドを使用してプロジェクトのソースをビルドします。

   c. すべてのプロジェクトで以下のコマンドを入力し、Maven を使用してプロジェクトで生成したすべてのアーティファクトのランタイムの依存関係をすべてダウンロードするようにします。

   ```
   mvn -e -DskipTests dependency:go-offline -f /path/to/project/pom.xml --batch-mode -
   Djava.net.preferIPv4Stack=true
   ```

   /path/to/project/pom.xml は、プロジェクトの **pom.xml** ファイルへの正しいパスに置き換えます。

   d. ローカルの Maven キャッシュディレクトリー (~/**.m2/repository**) のコンテンツを、使用中の一時ディレクトリーにコピーします。

3. 一時ディレクトリーのコンテンツを、Red Hat Decision Manager をデプロイしたコンピューターのディレクトリーにコピーします。このディレクトリーがオフラインの Maven ミラーリポジトリーになります。

4. 「Maven 設定ファイルの修正」 の説明に従い、Red Hat Decision Manager デプロイメント向けに、**settings.xml** ファイルを作成して設定します。

5. **settings.xml** ファイルで以下を変更します。

   - **<profile>** タグの下で、**<repositories>** タグまたは **<pluginRepositores>** タグがない場合には、必要に応じて追加します。

   - **<repositories>** の下に、以下のシーケンスを追加します。

   ```
   <repository>
     <id>offline-repository</id>
     <url>file:///path/to/repo</url>
     <releases>
       <enabled>true</enabled>
     </releases>
     <snapshots>
       <enabled>false</enabled>
     </snapshots>
   </repository>
   ```

   /path/to/repo は、ローカルの Maven ミラーリポジトリーのディレクトリーに対する完全パスに置き換えます。

   - **<pluginRepositories>** の下に、以下のシーケンスを追加します。

   ```
   <repository>
   ```

```
    <id>offline-plugin-repository</id>
    <url>file:///path/to/repo</url>
    <releases>
      <enabled>true</enabled>
    </releases>
    <snapshots>
      <enabled>false</enabled>
    </snapshots>
  </repository>
```

/path/to/repo は、ローカルの Maven ミラーリポジトリーのディレクトリーに対する完全パスに置き換えます。

第10章 GIT リポジトリーからのプロジェクトのインポート

Git は分散バージョン管理システムです。リビジョンをコミットオブジェクトとして実装します。リポジトリーに変更を保存すると、Git リポジトリーに新しいコミットオブジェクトが作成されます。

Business Central は Git を使用してプロジェクトデータ (ルールやプロセスなどのアセットを含む) を格納します。Business Central でプロジェクトを作成すると、Business Central に埋め込まれている Git リポジトリーに追加されます。他の Git リポジトリーにプロジェクトがある場合は、Business Central スペースから、そのプロジェクトを Business Central の Git リポジトリーにインポートできます。

前提条件

- Red Hat Decision Manager プロジェクトが外部の Git リポジトリーに存在している。
- 外部の Git リポジトリーへの読み取りアクセスに必要な認証情報がある。

手順

1. Business Central で **Menu** → **Design** → **Projects** の順にクリックします。
2. プロジェクトをインポートするスペースを選択または作成します。デフォルトのスペースは **mySpace** です。
3. 画面右側の縦に並んだ 3 つの点をクリックし、**Import Project** を選択します。
4. **Import Project** ウィンドウに、インポートするプロジェクトが含まれる Git リポジトリーの URL および認証情報を入力し、**Import** をクリックします。プロジェクトが Business Central の Git リポジトリーに追加され、現在のスペースで利用できるようになります。

第11章 LDAP と SSL の統合

Red Hat Decision Manager を使用して、RH-SSO を介して LDAP および SSL を統合できます。詳細は **Red Hat Single Sign-On サーバー管理ガイド** を参照してください。

付録A バージョン情報

本書の最終更新日: 2021 年 11 月 15 日 (月)