



## **Red Hat Decision Manager 7.0**

### **Red Hat Decision Manager のオンプレミスインストール**



# Red Hat Decision Manager 7.0 Red Hat Decision Manager のオンプレミス インストール

---

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

## 法律上の通知

Copyright © 2018 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, 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 Software Collections 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 Enterprise Application Platform 7.1 および Red Hat JBoss Web Server with Tomcat 8 上に Red Hat Decision Manager 7.0 をインストールして実行する方法を説明します。

## 目次

はじめに .....	4
<b>第1章 インストール計画 .....</b>	<b>5</b>
1.1. RED HAT DECISION MANAGER のコンポーネント	5
1.2. インストールオプション	5
1.3. ユーザーロール	6
1.4. 関連ツール	6
1.4.1. アセットリポジトリ	6
1.4.2. Apache Maven	6
1.4.3. Red Hat JBoss Developer Studio	7
1.4.4. Red Hat Business Optimizer	7
1.5. 高可用性とクラスター化	8
<b>第2章 RED HAT DECISION MANAGER のインストール .....</b>	<b>9</b>
2.1. RED HAT DECISION MANAGER インストールファイルのダウンロード	9
2.2. インストーラーを使用した RED HAT JBOSS EAP または RED HAT JBOSS WEB SERVE への RED HAT DECISION MANAGER のインストール	10
2.2.1. 対話モードのインストーラーを使用した Red Hat Decision Manager のインストール	11
2.2.2. CLI モードのインストーラーを使用した Red Hat Decision Manager のインストール	14
2.3. ZIP ファイルを使用した RED HAT JBOSS EAP への RED HAT DECISION MANAGER のインストール	15
2.3.1. デプロイ可能な ZIP ファイルを使用した Decision Central のインストール	15
2.3.2. デプロイ可能な ZIP ファイルを使用した Decision Server のインストール	16
2.3.3. ユーザーの作成	17
2.3.4. Decision Server の設定	18
2.3.5. Red Hat Decision Manager の実行	19
2.4. ZIP ファイルを使用した RED HAT JBOSS WEB SERVER への DECISION SERVER のインストール	20
2.4.1. Red Hat JBoss Web Server 上の Decision Server の確認	22
2.5. スタンドアロン DECISION CENTRAL の実行	22
2.5.1. サポートされるプロパティ	24
2.6. スタンドアロン DECISION SERVER CONTROLLER のインストールと実行	26
2.6.1. Red Hat JBoss EAP へのスタンドアロン Decision Server Controller のインストール	26
2.6.1.1. ユーザーの作成	27
2.6.1.2. Decision Server およびスタンドアロン Decision Server Controller の設定	27
2.6.1.3. スタンドアロン Decision Server Controller の実行	29
2.6.2. Red Hat JBoss Web Server へのスタンドアロン Decision Server Controller のインストール	30
<b>第3章 関連ツールのインストール .....</b>	<b>32</b>
3.1. プロジェクトでの MAVEN リポジトリの使用	32
3.1.1. プロジェクト設定ファイル (pom.xml) を使用した Maven の設定	32
3.1.2. 設定ファイルを使用した Maven の設定	32
3.1.3. Maven 依存関係の管理	33
3.2. GIT リポジトリからのプロジェクトのインポート	36
3.3. RED HAT JBOSS DEVELOPER STUDIO のインストールおよび設定	36
3.3.1. Red Hat JBoss Developer Studio プラグインのインストール	37
3.3.2. Red Hat Decision Manager サーバーの設定	37
3.3.3. Git リポジトリから Red Hat JBoss Developer Studio へのプロジェクトのインポート	38
3.3.4. リモート Git リポジトリのクローン作成	38
3.3.5. ローカル Git リポジトリのインポート	39
<b>第4章 RED HAT DECISION MANAGER におけるクラスター化 (デザインタイム開発環境向け) .....</b>	<b>40</b>
4.1. ELASTICSEARCH の設定	40
4.2. ACTIVEMQ JMS ブローカーのアクティブ化	41

4.3. クラスターノードへの DECISION CENTRAL のインストール	41
<b>第5章 RED HAT DECISION MANAGER システムの検証</b>	<b>43</b>
<b>第6章 DECISION CENTRAL のカスタマイズ</b>	<b>44</b>
6.1. DECISION CENTRAL ログインページのカスタマイズ	44
6.2. DECISION CENTRAL アプリケーションヘッダーのカスタマイズ	44
<b>第7章 RED HAT DECISION MANAGER と RED HAT SINGLE SIGN-ON の統合</b>	<b>45</b>
RH-SSO 統合ポイント	45
7.1. RH-SSO を使用した RED HAT DECISION MANAGER の認証	45
7.1.1. RH-SSO への Red Hat Decision Manager レルムクライアントの設定	45
7.1.2. Red Hat JBoss EAP の RH-SSO クライアントアダプターの設定	47
7.1.3. 新規ユーザーの追加	49
7.1.4. RH-SSO を使用した Red Hat Decision Manager リモートサービスのセキュリティ確保	49
7.1.5. RH-SSO を使用した Red Hat Decision Manager ファイルシステムサービスのセキュリティ確保	50
7.1.6. RH-SSO のユーザーおよびグループ管理の有効化	52
7.2. RH-SSO を使用した DECISION SERVER の認証	53
7.2.1. RH-SSO における Decision Server クライアントの作成	53
7.2.2. クライアントアダプターおよび Decision Server のインストールおよび設定	53
7.3. RH-SSO を使用したサードパーティークライアントの認証	56
7.3.1. 基本認証	56
7.3.2. トークンベースの認証	56
7.4. LDAP および SSL の RED HAT DECISION MANAGER との統合	57
<b>付録A バージョン情報</b>	<b>58</b>



## はじめに

Red Hat Decision Manager は、ビジネスルール管理と複合イベント処理を組み合わせるオープンソースの意思決定管理プラットフォームであり、意思決定管理機能およびビジネスリソースの最適化機能が含まれます。Red Hat Decision Manager を使用して、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。

Red Hat Decision Manager は、すべてのリソースが保存される集中リポジトリを使用します。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 担当者のサポートを受けることなくビジネスロジックおよびビジネスプロセスを編集することができます。



# 第1章 インストール計画

Red Hat Decision Manager のインストールを開始する前に、考慮しなければならないオプションが多数あります。本章の情報を確認し、インストールオプションおよび Red Hat Decision Manager と共に機能するコンポーネントの概要を把握してください。

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

Red Hat Decision Manager は、Decision Central、Decision Server、および Red Hat Business Optimizer で構成されます。

- Decision Central は、ビジネスルールを作成および管理するためのグラフィカルインターフェースです。
- Decision Server は、ルールおよびその他のアーティファクトが保存されるサーバーです。Decision Server を使用してルールのインスタンスを作成して実行し、プランニングの問題を解決します。
- Red Hat Business Optimizer は組み込み可能な軽量プランニングエンジンで、プランニングの問題を最適化します。

## 1.2. インストールオプション

ご自分の環境およびプロジェクトの要求に応じて、以下のインストールオプションのいずれかを選択します。



### 注記

- Decision Server を IBM WebSphere Application Server にインストールする方法については、『[IBM WebSphere Application Server への Decision Server のインストールおよび設定](#)』を参照してください。
- Decision Server を Oracle WebLogic Server にインストールする方法については、『[Oracle WebLogic Server への Decision Server のインストールおよび設定](#)』を参照してください。
- Red Hat Business Optimizer のインストールについては、『[Red Hat Business Optimizer のインストールおよび設定](#)』を参照してください。
- Red Hat JBoss EAP 7.1 または Red Hat JBoss Web Server 3.1 with Tomcat 8 上にインストールするための、実行可能な JAR インストーラーをダウンロードして実行します。インストーラーのグラフィカルインターフェースにより、インストールプロセスをステップごとに進めることができます。
- 以下のいずれかの ZIP ファイルをダウンロードする。ZIP ファイルによるインストールにグラフィカルインターフェースは必要ありません。したがって、SSH を使用して Red Hat Decision Manager をインストールできます。
  - Red Hat JBoss EAP 7.1 上に Red Hat Decision Manager をインストールするには、以下のファイルをダウンロードします。
    - `rhdm-7.0.0.GA-decision-central-eap7-deployable.zip`
    - `rhdm-7.0.0.GA-kie-server-ee7.zip`

- Red Hat JBoss Web Server 3.1 with Tomcat 8 上に Decision Server をインストールするには、**rhdm-7.0-kie-server-jws.zip** ファイルをダウンロードします。

詳細については、「[Red Hat Decision Manager 7 Supported Configurations](#)」を参照してください。

インストール手順については、「[2章 Red Hat Decision Manager のインストール](#)」を参照してください。

## 1.3. ユーザーロール

Red Hat Decision Manager では、以下のユーザーロールを利用できます。

- **admin: admin** ロールを持つユーザーは、Red Hat Decision Manager の管理者です。管理者は、ユーザーの管理やリポジトリの管理、作成、およびクローンが可能で、アプリケーションに必要な変更を加えるためのあらゆるアクセス権限を持ちます。**admin** ロールを持つユーザーは、システム内のすべてのエリアにアクセスできます。Decision Central を使用する前に、**admin** ロールを持つユーザーを作成する必要があります。
- **analyst: analyst** ロールを持つユーザーは、プロジェクトをモデル化するためのすべての上位機能にアクセスできますが、このユーザーは **Authoring** → **Administration** にアクセスすることはできません。また、**Deployment** → **Artifact Repository** ビューなど、開発者を対象とする下位機能の一部にもアクセスすることはできませんが、**analyst** ロールを持つユーザーは Project Editor の使用時に **Build & Deploy** ボタンを使用できます。
- **kie-server: kie-server** ロールを持つユーザーは、Decision Server (KIE Server) の REST 機能にアクセスできます。Decision Server にログインするには、**kie-server** ロールを持つユーザーを作成する必要があります。
- **rest-all: rest-all** ロールを持つユーザーは、Decision Central の REST 機能にアクセスできます。

詳細については、「[ユーザーの作成](#)」を参照してください。

## 1.4. 関連ツール

アセットリポジトリ、Apache Maven、Red Hat JBoss Developer Studio、および Red Hat Business Optimizer は、Red Hat Decision Manager と統合することでそれぞれ重要な機能を果たします。

### 1.4.1. アセットリポジトリ

Decision Central で作成されるビジネスルールならびにその他のアセットおよびリソースは、アセットリポジトリに保存されます。アセットリポジトリは、ナレッジストアとしても知られています。

ナレッジストアは、ビジネスナレッジを保存する集中リポジトリのことです。ナレッジストアは Git リポジトリと連携し、さまざまなナレッジアセットやアーティファクトを 1 箇所で保存します。Decision Central は、保存されたコンテンツを表示、更新できる Web フロントエンドとして機能します。コンテンツへは、Red Hat Decision Manager の一元化された環境から Project Explorer を使用してアクセスできます。

すべてのビジネスアセットは、リポジトリに保存されます。

### 1.4.2. Apache Maven

[Apache Maven](#) は分散型構築自動化ツールで、ソフトウェアプロジェクトのビルドおよび管理を行うた

めに Java アプリケーション開発で使用されます。Maven を使用して、ご自分の Red Hat Decision Manager プロジェクトをビルド、公開、およびデプロイすることができます。Maven には以下のメリットがあります。

- ビルドプロセスが容易で、すべてのプロジェクトに対して統一された構築システムが実装される。
- プロジェクトに必要なすべての JAR ファイルがコンパイル時に利用可能になる。
- 適切なプロジェクト構造が設定される。
- 依存関係およびバージョンが適切に管理される。
- Maven では事前定義されたさまざまな出力タイプ (JAR および WAR 等) にビルドされるため、追加のビルドプロセスが不要である。

Maven はリポジトリを使用して Java ライブラリー、プラグイン、およびその他のビルドアーティファクトを保存します。これらのリポジトリは、ローカルまたはリモートいずれかの形態をとることができます。Red Hat Decision Manager によりローカルおよびリモート maven リポジトリが維持され、それをご自分のプロジェクトに追加してルール、プロセス、イベント、およびその他のプロジェクト依存関係にアクセスできます。プロジェクトおよびアーキタイプをビルドする際に、Maven はローカルまたはリモートリポジトリから Java ライブラリーおよび Maven プラグインを動的に取得します。これにより、プロジェクト全体を通じて依存関係の共有および再利用が促進されます。

Apache Maven の設定方法については、[「プロジェクトでの Maven リポジトリの使用」](#)を参照してください。

### 1.4.3. Red Hat JBoss Developer Studio

Red Hat JBoss Developer Studio は Eclipse をベースにした統合開発環境 (IDE) です。Eclipse、Eclipse Tooling、および Red Hat JBoss EAP を組み合わせることで、ツールとランタイムコンポーネントを統合します。Red Hat JBoss Developer Studio は Red Hat Decision Manager 用のツールおよびインターフェースを持つプラグインを提供します。これらのプラグインはコミュニティバージョンの製品が基になっています。そのため、Red Hat Decision Manager プラグインは Drools プラグインと呼ばれます。

Red Hat JBoss Developer Studio の詳細については、[「Red Hat JBoss Developer Studio のインストールおよび設定」](#)を参照してください。

### 1.4.4. Red Hat Business Optimizer

Red Hat Business Optimizer は組み込み可能な軽量プランニングエンジンで、プランニングの問題を最適化します。最適化のためのヒューリスティック法およびメタヒューリスティック法を効率的なスコア計算と組み合わせ、Java プログラマーがプランニングの問題を効率的に解決できるようにします。

Red Hat Business Optimizer は、さまざまなユースケースの解決に役立ちます。以下にその例を示します。

- **従業員勤務表/患者一覧:** 看護師の勤務シフト作成を容易にし、病床管理を追跡します。
- **教育機関の時間割:** 授業、コース、試験、および会議の計画を容易にします。
- **工場の計画:** 自動車の組み立てライン、機械の待機計画、および作業員のタスク計画を追跡します。
- **在庫の削減:** 紙や金属などの資源の消費を削減し、無駄を最小限に抑えます。

どの組織も、プランニングの問題に直面しています。彼らは制約のある限定されたリソース (従業員、資産、時間、資金) を使用して製品やサービスを提供しています。Red Hat Business Optimizer は、最適化のためのヒューリスティック法およびメタヒューリスティック法を効率的なスコア計算と組み合わせ、Java プログラマーが制約充足問題を効率的に解くことができますようにします。

詳細については、[『Red Hat Business Optimizer のインストールおよび設定』](#)を参照してください。

## 1.5. 高可用性とクラスター化

高可用性とは、必要な期間継続して稼働する、または利用可能なシステムまたはコンポーネントのことを指します。可用性は、事実上不可能な「100% 稼働」または「機能停止なし」に対する相対値として測定できます。一般的ではあるが実現するのが困難なシステム/製品向けの可用性の指標は、「ファイブナイン」(99.999%) の可用性として知られています。

高可用性 (HA) クラスターはサービスのグループを指し、これを使用してダウンタイムが最小限、またはダウンタイムがないシステムを構築できます。クラスター化していないと、サービスがクラッシュした場合や負荷が非常に高い場合に、そのサービスを要求しているユーザーがすぐに応答を得ることができません。高可用性クラスター構成では、複数のノードがデータのコピーおよびサービスを提供します。サービスのウォッチドッグがクラスター内にあるノードの異常を検出すると、異常の発生したノードを再起動すると同時に、サービスを別のノードに切り替えます。多くの場合、ユーザーが異常に気付くことはありません。

Red Hat Decision Manager で高可用性クラスターを設定する方法については、「[4章 Red Hat Decision Manager におけるクラスター化 \(デザインタイム開発環境向け\)](#)」を参照してください。

## 第2章 RED HAT DECISION MANAGER のインストール

Red Hat JBoss Enterprise Application Platform (Red Hat JBoss EAP) 7.1 は、Java Enterprise Edition 7 (Java EE 7) の Full Profile および Web Profile 仕様の認定実装です。Red Hat JBoss EAP には、高可用性クラスタリング、メッセージング、および分散キャッシング等の機能に対する事前設定オプションが用意されています。ユーザーは、Red Hat JBoss EAP の提供するさまざまな API およびサービスを使用して、アプリケーションを開発、デプロイ、および実行することもできます。

Red Hat JBoss Web Server は Tomcat 8 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/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 Decision Manager は、Decision Central および Decision Server で構成されます。Decision Central は、Red Hat JBoss EAP 7.1 サーバースystemにインストールできます。Decision Server は、Red Hat JBoss EAP 7.1 または Red Hat JBoss Web Server システムにインストールできます。



### 注記

Decision Server を IBM WebSphere Application Server および Oracle Weblogic Server にインストールすることもできます。詳細については、以下のドキュメントを参照してください。

- [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](#)

本書の手順では、Red Hat Decision Manager をオンプレミスの Red Hat JBoss EAP 7.1 および Red Hat JBoss Web Server 3.1 上にインストールする方法について説明します。Red Hat OpenShift Container Platform へのインストールについては、『[Red Hat OpenShift Container Platform への Red Hat Decision Manager のデプロイメント](#)』を参照してください。

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 RHDM product and maven library version?](#)
- [Red Hat Decision Manager 7 Supported Configurations](#)



### 注記

本セクションでは、Decision Central と Decision Server を同じサーバー上にインストールする手順を説明します。Red Hat では、実稼働環境では Decision Central と Decision Server を異なるサーバー上にインストールすることを推奨します。

## 2.1. RED HAT DECISION MANAGER インストールファイルのダウンロード

ご自分の環境およびインストール要求に応じた 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](#) にログインします。

2. ページ上部の **ダウンロード** をクリックします。
3. 表示された「**製品のダウンロード**」ページで JBoss 統合および自動化 セクションに移動し、**Red Hat Decision Manager** をクリックします。
4. 「**Software Downloads**」ページにおいて、必要に応じて **Product** メニューから **Decision Manager** を選択し **Version** メニューから **7.0** を選択します。
5. 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。
  - インストーラーを使用して Red Hat JBoss EAP 7.1 上に Red Hat Decision Manager をインストールする、または Red Hat JBoss Web Server 3.1 上に Decision Server をインストールするには、**Red Hat Decision Manager 7.0.0 Installer (rhdm-installer-7.0.0.GA.jar)** をダウンロードします。
  - デプロイ可能な zip ファイルを使用して Red Hat JBoss EAP 7.1 上に Red Hat Decision Manager をインストールするには、以下のファイルをダウンロードします。
    - **Red Hat Decision Manager 7.0.0 Decision Server for All Supported EE7 Containers (rhdm-7.0.0.GA-kie-server-ee7.zip)**
    - **Red Hat Decision Manager 7.0.0 Decision Central Deployable for EAP 7 (rhdm-7.0.0.GA-kie-server-ee7.zip)**
  - アプリケーションサーバーにデプロイせずに Decision Central を実行するには、**Red Hat Decision Manager 7.0.0 Decision Central Standalone (rhdm-7.0.0.GA-decision-central-standalone.jar)** をダウンロードします。
  - デプロイ可能な zip ファイルを使用して Red Hat JBoss Web Server 3.1 上に Decision Server をインストールするには、**Red Hat Decision Manager 7.0.0 Add-On's (rhdm-7.0.0.GA-add-ons.zip)** をダウンロードします。

## 次のステップ

以下のセクションのいずれかに移動します。

- [「インストーラーを使用した Red Hat JBoss EAP または Red Hat JBoss Web Serve への Red Hat Decision Manager のインストール」](#)
- [「ZIP ファイルを使用した Red Hat JBoss EAP への Red Hat Decision Manager のインストール」](#)
- [「スタンドアロン Decision Central の実行」](#)

## 2.2. インストーラーを使用した RED HAT JBOSS EAP または RED HAT JBOSS WEB SERVE への RED HAT DECISION MANAGER のインストール

本セクションでは、インストーラー JAR ファイルを使用して Red Hat Decision Manager をインストールするのに必要な手順を説明します。JAR ファイルは、既存の Red Hat JBoss EAP 7.1 または Red Hat JBoss Web Server 3.1 with Tomcat 8 サーバーシステムに Red Hat Decision Manager をインストールする実行ファイルです。標準モードまたはコマンドラインインターフェース (CLI) モードで、インストーラーを実行できます。



**注記**

Red Hat Decision Manager JAR ファイルインストーラーは、yum または RPM パッケージマネージャーでインストールした Red Hat JBoss EAP ディストリビューションをサポートしません。このような Red Hat JBoss EAP システムに Red Hat Decision Manager をインストールする場合は、Red Hat Decision Manager 7.0 Deployable for Red Hat JBoss EAP 7.1 ファイルをダウンロードして、「[ZIP ファイルを使用した Red Hat JBoss EAP への Red Hat Decision Manager のインストール](#)」に記載の手順に従います。

**注記**

IBM JDK は他の JDK で生成されるキーストアを使用できないため、別の JDK で生成されたキーストアを持つ IBM JDK で実行されている既存の Red Hat JBoss EAP システムに Red Hat Decision Manager をインストールすることはできません。

**次のステップ:**

以下のセクションのいずれかに記載された手順に従います。

- [「対話モードのインストーラーを使用した Red Hat Decision Manager のインストール」](#)
- [「CLI モードのインストーラーを使用した Red Hat Decision Manager のインストール」](#)

### 2.2.1. 対話モードのインストーラーを使用した **Red Hat Decision Manager** のインストール

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。このファイルを使用して、既存の Red Hat JBoss EAP 7.1 または Red Hat JBoss Server 3.1 with Tomcat 8 サーバーシステムに Red Hat Decision Manager をインストールできます。

**注記**

セキュリティ上の理由で、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 EAP 7.1 (もしくはそれ以降) または Red Hat JBoss Web Server 3.1 with Tomcat 8 (もしくはそれ以降) サーバーシステム
- インストールを完了するのに十分なユーザーパーミッション

**注記**

Decision Server を Red Hat JBoss Web Server 3.1 上にインストールする場合は、必ず Tomcat 8 への書き込み権限を持った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 を使用するようには、**"-Dfile.encoding=UTF-8"** のシステムプロパティを使用します。

## 手順

1. ターミナルウィンドウにおいて、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に移動し、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.0.0.GA.jar
```



## 注記

Windows でインストーラーを実行する場合には、インストール時に管理者の認証情報を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-7.0.0.GA.jar
```

また、32 ビットの Java 仮想マシン上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-7.0.0.GA.jar
```

2. 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。
3. **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。
4. Red Hat Decision Manager をインストールする Red Hat JBoss EAP 7.1 または Red Hat JBoss 3.1 Web Server with Tomcat 8 サーバーのホームを指定し、**Next** をクリックします。
5. 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。  
Decision Central を Red Hat JBoss 3.1 Web Server with Tomcat 8 上にインストールすることはできません。Decision Central をインストールできるのは Red Hat JBoss EAP だけです。ただし、Red Hat JBoss 3.1 Web Server with Tomcat 8 上に Decision Server コントローラーをインストールすることができます。コントローラーは、Decision Server を管理するために使用されます。複数の Decision Server インスタンスを管理する場合には、コントローラーをインストールしてください。



## 注記

Decision Central と Decision Server を同じサーバー上にインストールすることは可能です。ただし Red Hat では、実稼働環境では Decision Central と Decision Server を異なるサーバー上にインストールすることを推奨します。そのためには、インストーラーを 2 回実行します。



6. Decision Manager の **admin** ユーザーを作成し、**Next** をクリックします。



### 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選択してください。たとえば、**admin** という名前のユーザーを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。Decision Central および Decision Server にアクセスする際に必要になります。

7. Component Installation ページで **Next** をクリックし、インストールを開始します。Component Installation ページに、インストールされるコンポーネントのリストが表示されます。
8. インストールが完了したら、**Processing Finished** ページで **Next** をクリックします。次のページに **Installation has completed successfully** というメッセージが表示されます。
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実施するには、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.0.0.GA.jar <path-to-file>
```

インストーラーを使用して、正常に Red Hat Decision Manager をインストールすることができました。Red Hat JBoss EAP 上に Decision Central だけをインストールした場合は、これらのステップを繰り返して別のサーバーに Decision Server をインストールします。

Red Hat Decision Manager を起動するには、以下のコマンドのいずれかを入力します。

- Red Hat JBoss EAP 7.1 上にインストールした場合は、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:

```
$ EAP_HOME/bin/standalone.sh
```

- Windows の場合:

```
EAP_HOME\bin\standalone.bat
```

- Red Hat JBoss Web Server 3.1 with Tomcat 8 上にインストールした場合には、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:

```
JWS_HOME/bin/startup.sh
```

- Windows の場合:

-

```
JWS_HOME\bin\startup.sh
```

### 2.2.2. CLI モードのインストーラーを使用した Red Hat Decision Manager のインストール

コマンドラインインターフェース (CLI) を使用して Red Hat Decision Manager のインストーラーを実行することができます。



#### 注記

セキュリティ上の理由で、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

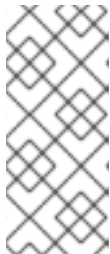
- バックアップを作成してある Red Hat JBoss EAP 7.1 (もしくはそれ以降) または Red Hat JBoss Web Server 3.1 with Tomcat 8 (もしくはそれ以降) サーバーシステム
- インストールを完了するのに十分なユーザーパーミッション



#### 注記

Decision Server を Red Hat JBoss Web Server 3.1 上にインストールする場合は、必ず Tomcat 8 への書き込み権限を持った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 を使用するようには、`"-Dfile.encoding=UTF-8"` のシステムプロパティを使用します。

#### 手順

1. ターミナルウィンドウにおいて、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-7.0.0.GA.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から **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 7.1 or Tomcat 8
[/home/user/RHDM-7.0.0/jboss-eap-7.1]
```

3. 既存の Red Hat JBoss EAP 7.1 または Red Hat JBoss Web Server 3.1 with Tomcat 8 システム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。



#### 注記

Decision Manager と Decision Server を同じサーバー上にインストールすることは可能です。ただし Red Hat では、実稼働環境では Decision Central と Decision Server を異なるサーバー上にインストールすることを推奨します。

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、指定したユーザー名が既存のロールタイトルまたはグループ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、**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。Decision 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。

正常に Red Hat Decision Manager をインストールすることができました。Decision Central だけをインストールした場合は、これらのステップを繰り返して別のサーバーに Decision Server をインストールします。

## 2.3. ZIP ファイルを使用した RED HAT JBOSS EAP への RED HAT DECISION MANAGER のインストール

Red Hat JBoss EAP 上に Red Hat Decision Manager 7.0 をインストールするためのデプロイ可能な ZIP ファイルは、2 つの ZIP ファイルで構成されます。1 つは Decision Central 用で、もう 1 つは Decision Server 用です。

実稼働環境では、Decision Central と Decision Server を異なるサーバー上にインストールする必要があります。

### 2.3.1. デプロイ可能な ZIP ファイルを使用した Decision Central のインストール

Decision Central の Web コンソールにより、個々のコンポーネントに対する以下のタスクを一元化された Web ベースの環境で実施することができます。

- ルールおよび関連アセットを作成、管理、および編集する。
- 接続された Decision Server およびそのコンテナを管理する。

#### 前提条件

- バックアップを作成してある Red Hat JBoss EAP システム (バージョン 7.1 またはそれ以降)
  - インストールを完了するのに十分なユーザーパーミッション
  - 以下のファイル ([Red Hat カスタマーポータル](#) の「Software Downloads」ページにアクセスし、**Decision Central Deployable for EAP 7** の横の **Download** をクリックしてダウンロードすることができます)
- rhdm-7.0.0.GA-decision-central-eap7-deployable.zip**

#### 手順

1. **rhdm-7.0.0.GA-decision-central-eap7-deployable.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** としています。
2. **TEMP\_DIR/rhdm-7.0.0.GA-decision-central-eap7-deployable/jboss-eap-7.1** ディレクトリーの内容を **EAP\_HOME** にコピーします。ファイルの上書きまたはディレクトリーのマージを確認されたら、**はい** を選択します。



#### 警告

コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss EAP 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。

### 2.3.2. デプロイ可能な ZIP ファイルを使用した 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 (ナレッジストア) に保存されたデータにアクセスします。

#### 前提条件

- バックアップを作成してある Red Hat JBoss EAP システム (バージョン 7.1 またはそれ以降)。Red Hat JBoss EAP システムのベースディレクトリーは、**EAP\_HOME** と表示しています。
  - インストールを完了するのに十分なユーザーパーミッション
  - 以下のファイル ([Red Hat カスタマーポータル](#) の「Software Downloads」ページにアクセスし、**Decision Server for All Supported EE7 Containers** の横の **Download** をクリックしてダウンロードすることができます)
- rhdm-7.0.0.GA-kie-server-ee7.zip**

## 手順

1. **rhdm-7.0.0.GA-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** としています。
2. **TEMP\_DIR/rhdm-7.0.0.GA-kie-server-ee7/rhdm-7.0.0.GA-kie-server-ee7/kie-server.war** ディレクトリーを **EAP\_HOME/standalone/deployments/** にコピーします。



## 警告

コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss EAP 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。

3. **TEMP\_DIR/rhdm-7.0.0.GA-kie-server-ee7/rhdm-7.0.0.GA-kie-server-ee7/SecurityPolicy/** の内容を **EAP\_HOME/bin** にコピーします。ファイルの上書きを確認されたら、**はい** を選択します。
4. **EAP\_HOME/standalone/deployments/** ディレクトリーに、**kie-server.war.dodeploy** という名前で空のファイルを作成します。このファイルにより、サーバーが起動すると Decision Server が自動的にデプロイされます。

## 2.3.3. ユーザーの作成

Decision Central を使用する前に、**admin** ロールを持つユーザーを作成する必要があります。Decision Server にログインするには、**kie-server** ロールを持つユーザーを作成する必要があります。1 人のユーザーに両方のロールを持たせることもできます。

## 前提条件

以下のセクションのいずれかに記載された手順に従って **EAP\_HOME** にインストールされた Red Hat Decision Manager

- 「対話モードのインストーラーを使用した Red Hat Decision Manager のインストール」
- 「CLI モードのインストーラーを使用した Red Hat Decision Manager のインストール」
- 「ZIP ファイルを使用した Red Hat JBoss EAP への Red Hat Decision Manager のインストール」

## 手順

1. ターミナルアプリケーションで **EAP\_HOME/bin** ディレクトリーに移動します。
2. Decision Central および Decision Server へのログインに使用するユーザーを作成します。以下のコマンドの **<username>** および **<password>** を、ご自分で考えたユーザー名およびパスワードに置き換えます。

```
$ ./add-user.sh -a --user <USERNAME> --password <PASSWORD> --role
kie-server,admin
```



### 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を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。

3. ユーザー名とパスワードを書き留めておきます。

## 2.3.4. Decision Server の設定

Decision Server が Decision Central により管理される場合は、本セクションの説明に従って Decision Server および Decision Central システム両方の **standalone.xml** ファイルを編集する必要があります。



### 注記

これらの変更を行うのは、Decision Server が Decision Central により管理される場合だけです。

### 前提条件

- 以下のセクションのいずれかに記載された手順に従って **EAP\_HOME** にインストールされた Red Hat Decision Manager
  - 「対話モードのインストーラーを使用した Red Hat Decision Manager のインストール」
  - 「CLI モードのインストーラーを使用した Red Hat Decision Manager のインストール」
  - 「ZIP ファイルを使用した Red Hat JBoss EAP への Red Hat Decision Manager のインストール」



### 注記

Red Hat では、実稼働環境では Decision Server と Decision Central を異なるサーバー上にインストールすることを推奨します。ただし、Decision Server と Decision Central を同じサーバー上にインストールする場合には (たとえば開発環境として)、共有する **standalone.xml** ファイルに対してこれらの変更を行います。

- Decision Server ノード: **kie-server** ロールを持つユーザー
- Decision Central サーバーノード: **rest-all** ロールを持つユーザー  
詳細については、「[ユーザーの作成](#)」を参照してください。

### 手順

1. Decision Central の **EAP\_HOME/standalone/configuration/standalone.xml** ファイルにおいて、**<system-properties>** セクションの以下のプロパティのコメントを解除し、**<USERNAME>** および **<USER\_PWD>** を **kie-server** ロールを持つユーザーの認証情報に置

き換えます。

```
<property name="org.kie.server.user" value="<USERNAME>"/>
<property name="org.kie.server.pwd" value="<USER_PWD>"/>
```

- Decision Server の **EAP\_HOME/standalone/configuration/standalone.xml** ファイルにおいて、**<system-properties>** セクションの以下のプロパティーのコメントを解除します。

```
<property name="org.kie.server.controller.user" value="
<CONTROLLER_USER>"/>
<property name="org.kie.server.controller.pwd" value="
<CONTROLLER_PWD>"/>
<property name="org.kie.server.id" value="<KIE_SERVER_ID>"/>
<property name="org.kie.server.location" value="http://<HOST>:
<PORT>/kie-server/services/rest/server"/>
<property name="org.kie.server.controller" value="
<CONTROLLER_URL>"/>
```

- このファイルで、以下の値を置き換えます。

- **<CONTROLLER\_USER>** および **<CONTROLLER\_PWD>** を **rest-all** ロールを持つユーザーの認証情報に置き換える。
- **<KIE\_SERVER\_ID>** を Decision Server システムの ID または名前に置き換える (例: **rhdm700-decision-server-1**)。
- **<HOST>** を Decision Server ホストの ID または名前に置き換える (例: **localhost** または **192.7.8.9**)。
- **<PORT>** を Decision Server ホストのポートに置き換える (例: **8080**)。



#### 注記

**org.kie.server.location** プロパティーで Decision Server の場所を指定します。

- **<CONTROLLER\_URL>** を Decision Central に埋め込まれたコントローラーの URL に置き換える。Decision Server は、起動時にこの URL に接続されます。
  - インストーラーまたは Red Hat JBoss EAP zip ファイルを使用して Decision Central をインストールした場合、**<CONTROLLER\_URL>** は以下のフォーマットになります。  
<http://<HOST>:<PORT>/decision-central/rest/controller>
  - **standalone.jar** ファイルを使用して Decision Central を実行している場合、**<CONTROLLER\_URL>** は以下のフォーマットになります。  
<http://<HOST>:<PORT>/rest/controller>

### 2.3.5. Red Hat Decision Manager の実行

Red Hat JBoss EAP 上に Red Hat Decision Manager をインストールしたら、以下の手順を使用してスタンドアロンモードで Red Hat Decision Manager を実行します。

## 前提条件

- 「ZIP ファイルを使用した Red Hat JBoss EAP への Red Hat Decision Manager のインストール」に記載の手順に従って **EAP\_HOME** にインストールされた Red Hat Decision Manager
- 「Decision Server の設定」に記載の手順に従って設定された Red Hat Decision Manager

## 手順

1. ターミナルアプリケーションで **EAP\_HOME/bin** に移動します。
2. スタンドアロン設定を実行します。

- Linux または UNIX ベースのシステムの場合:

```
$ ./standalone.sh
```

- Windows の場合:

```
standalone.bat
```

### 注記

Red Hat JBoss EAP 上に Decision Central をデプロイせず Decision Server だけをデプロイした場合は、以下のコマンドのいずれかを使用して Red Hat JBoss EAP をスタンドアロン Full Profile で起動する必要があります。

Linux または UNIX ベースのシステムの場合:

```
$ /standalone.sh -c standalone-full.xml
```

Windows の場合:

```
standalone.bat -c standalone-full.xml
```

3. Web ブラウザーで URL **localhost:8080/decision-central** にアクセスします。
4. ユーザー名 **rhdmAdmin** およびパスワード **password@1** を使用してログインします。

## 2.4. ZIP ファイルを使用した RED HAT JBOSS WEB SERVER への 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 (ナレッジストア) に保存されたデータにアクセスします。本セクションでは、ZIP ファイルを使用して既存の Red Hat JBoss Web Server 3.1 インスタンス上に Decision Server をインストールする方法について説明します。

### 注記

インストーラー JAR ファイルを使用した Decision Server のインストールについては、「インストーラーを使用した Red Hat JBoss EAP または Red Hat JBoss Web Server への Red Hat Decision Manager のインストール」を参照してください。



## 前提条件

- バックアップを作成してある Red Hat JBoss Web Server 3.1 with Tomcat 8 (またはそれ以降の) サーバシステム。JBoss Web Server システムのベースディレクトリーは、**JWS\_HOME** と表示しています。
- インストールを完了するのに十分なユーザーパーミッション
- **rhdm-7.0-kie-server-jws.zip** ファイル。このファイルをダウンロードするには、以下の手順を実施します。
  - a. [Red Hat カスタマーポータル](#) の「Software Downloads」ページで、**Red Hat Decision Manager 7.0.0 Add-On's** の横の **Download** をクリックします。
  - b. **rhdm-7.0.0.GA-add-ons.zip** ファイルを展開します。**rhdm-7.0-kie-server-jws.zip** ファイルは、展開したディレクトリーにあります。

## 手順

1. **rhdm-7.0-kie-server-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、このディレクトリーを **TEMP\_DIR** としています。
2. **TEMP\_DIR/rhdm-7.0-kie-server-jws/kie-server.war** ディレクトリーを **JWS\_HOME/tomcat8/webapps** ディレクトリーにコピーします。



### 警告

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。

3. **kie-server.war** フォルダーから **.war** 拡張子を削除します。
4. テキストエディターで **JWS\_HOME/tomcat8/conf/tomcat-users.xml** ファイルを開きます。
5. **JWS\_HOME/tomcat8/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。サポートされるロールのリストについては、「[ユーザーロール](#)」を参照してください。**<USERNAME>** と **<PASSWORD>** は、ご自分で考えたユーザーとパスワードの組み合わせです。

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

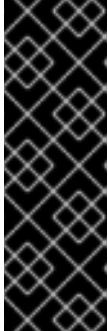
ユーザーに複数のロールを割り当てる場合は、以下の例に示すようにロールをコンマで区切ります。

```
<role rolename="admin"
```

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

6. **JWS\_HOME/tomcat8/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。

```
CATALINA_OPTS="-Xmx1024m -Dorg.jboss.server.ext.disabled=true -
Dorg.jboss.ui.server.ext.disabled=true -
Dorg.jboss.case.server.ext.disabled=true"
```



### 重要

Microsoft Windows の場合は、以下の値を **setenv.bat** ファイルに追加します。

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

7. JBoss Web Server を起動するには、**JWS\_HOME/tomcat8/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
```

8. 数分後に **JWS\_HOME/tomcat8/logs** ディレクトリーを確認し、エラーが発生していたら修正します。

## 2.4.1. Red Hat JBoss Web Server 上の Decision Server の確認

Decision Server が Red Hat JBoss Web Server 上で動作していることを確認するには、以下のコマンドを入力します。

```
curl -X GET "http://localhost:8080/kie-server/services/rest/server" -H
"accept: application/xml" -u '<CONTROLLER>:<CONTROLLER_PWD>'
```

このコマンドで、**<CONTROLLER>** および **<CONTROLLER\_PWD>** を **tomcat-users.xml** ファイル内の値に置き換えます。

このコマンドにより、Decision Server インスタンスに関する情報が出力されます。

## 2.5. スタンドアロン DECISION CENTRAL の実行

Decision Central スタンドアロン JAR ファイルを使用して、Red Hat JBoss EAP 等のアプリケーションサーバーにデプロイせずに Decision Central を実行することができます。



## 注記

Red Hat がこのタイプのインストールをサポートするのは、オンプレミスの Red Hat Enterprise Linux にインストールされた場合に限りです。

## 手順

1. [Red Hat カスタマーポータル](#) から Decision Central スタンドアロン JAR ファイルをダウンロードします。
2. ターミナルウィンドウにおいて、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ます。
3. 以下の内容で **application-config.yaml** 設定ファイルを作成します。

```
swarm:
  management:
    security-realms:
      ApplicationRealm:
        local-authentication:
          default-user: local
          allowed-users: local
          skip-group-loading: true
        properties-authentication:
          path: /path/to/application-users.properties
          plain-text: true
        properties-authorization:
          path: /path/to/application-roles.properties
datasource:
  management:
    wildfly:
      admin: admin
```

4. **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 Decision Central インスタンスが Decision Server のコントローラーになる場合は、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
```

5. **application-roles.properties** ファイルを作成し、**application-users.properties** ファイルに含めた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
```

詳細については、「[ユーザーロール](#)」を参照してください。

6. 以下のコマンドを入力します。

```
java -jar rhdm-7.0.0.GA-decision-central-standalone.jar -s
application-config.yaml
```

また、このコマンドに **-D<property>=<value>** パラメーターを含めて、Decision Central で

サポートされる任意のプロパティを設定することもできます。以下に例を示します。

```
java -jar rhdm-7.0.0.GA-decision-central-standalone.jar -s
application-config.yaml -D<property>=<value> -D<property>=<value>
```

詳細については、「[サポートされるプロパティ](#)」を参照してください。

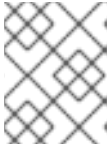
### 2.5.1. サポートされるプロパティ

スタンドアロン Decision Central をインストールする場合は、以下のコマンドにおいて本セクションに列記したプロパティを使用することができます。

```
java -jar rhdm-7.0.0.GA-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のいずれかで、**<value>** がそのプロパティに割り当てる値です。

- **org.uberfire.nio.git.dir**: ディレクトリー **.niogit** の場所。デフォルト: 作業ディレクトリー
- **org.uberfire.nio.git.dirname**: Git ディレクトリーの名前。デフォルト: **.niogit**
- **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: **true**
- **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: **localhost**
- **org.uberfire.nio.git.daemon.port**: Git デーモンが有効な場合は、このプロパティをポート番号として使用します。デフォルト: **9418**
- **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: **true**
- **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: **localhost**
- **org.uberfire.nio.git.ssh.port**: SSH デーモンが有効な場合は、このプロパティをポート番号として使用します。デフォルト: **8001**
- **org.uberfire.nio.git.ssh.cert.dir**: ローカルの証明書が保存されるディレクトリー **.security** 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 (例: [git@github.com:user/repository.git](#)) を持つ Git リポジトリ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。
- **org.uberfire.nio.git.ssh.algorithm**: SSH で使用されるアルゴリズム。デフォルト: **DSA**



## 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てください。

- **org.uberfire.metadata.index.dir**: Lucene .index フォルダ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ために使用される regex パターン。プリンシパルの値をロール名に一致させるときにアプリケーションのロール名で置き換えられるので、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**: 設定の監視を無効にします (内容を熟知していない限り、無効にしないでください)。デフォルト: false
- **org.uberfire.secure.key**: パスワード暗号化で使用するシークレットパスワード。デフォルト: org.uberfire.admin
- **org.uberfire.secure.alg**: パスワード暗号化で使用する暗号化アルゴリズム。デフォルト: PBESWithMD5AndDES
- **org.uberfire.domain**: uberfire により使用される security-domain 名。デフォルト: ApplicationRealm
- **org.guvnor.m2repo.dir**: Maven リポジトリフォルダーが保存される場所。デフォルト: working-directory/repositories/kie
- **org.guvnor.project.gav.check.disabled**: GAV チェックを無効にします。デフォルト: false
- **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: false
- **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: false
- **org.kie.server.controller**: Kie Server Controller との接続に使用される URL (例: `ws://localhost:8080/decision-central/websocket/controller`)。
- **org.kie.example**: GitHub 外部へのデモアプリケーションのクローン作成を有効にします。
- **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: false
- **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: false
- **org.kie.server.controller**: Kie Server Controller との接続に使用される URL (例: `ws://localhost:8080/decision-central/websocket/controller`)。
- **org.kie.server.user**: 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。このプロパティが必要なのは、この Decision Central システムをコントローラーとして使用する場合だけです。

- **org.kie.server.pwd**: 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パスワード。このプロパティが必要なのは、この Decision Central システムをコントローラーとして使用する場合だけです。

## 2.6. スタンドアロン **DECISION SERVER CONTROLLER** のインストールと実行

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。Decision Server が非管理モードにある場合は、手動でコンテナを作成および維持する必要があります。Decision Server が管理モードにある場合は、スタンドアロン Decision Server Controller が Decision Server の設定を管理し、ユーザーは Controller を操作してコンテナを作成および維持します。

スタンドアロン Decision Server Controller は、Decision Central に組み込まれています。Decision Central をインストールしている場合は、**Execution Server** ページを使用してコンテナを作成および維持します。Decision Central をインストールしていない場合は、スタンドアロン Decision Server Controller をインストールし、REST API または Decision Server Java Client API を使用して Controller を操作できます。

### 2.6.1. Red Hat JBoss EAP へのスタンドアロン **Decision Server Controller** のインストール

スタンドアロン Decision Server Controller をインストールし、REST API または Decision Server Java Client API を使用して Controller を操作できます。

#### 前提条件

- バックアップを作成してある Red Hat JBoss EAP システム (バージョン 7.1 またはそれ以降)。Red Hat JBoss EAP システムのベースディレクトリーは、**EAP\_HOME** と表示しています。
- インストールを完了するのに十分なユーザーパーミッション

#### 手順

1. [Red Hat カスタマーポータル](#) の「Software Downloads」ページで **Red Hat Decision Manager 7.0.0 Add-On's** の横の **Download** をクリックし、**rhdm-7.0-controller-ee7.zip** ファイルをダウンロードします。
2. **rhdm-7.0.0.GA-add-ons.zip** ファイルを展開します。**rhdm-7.0-controller-ee7.zip** ファイルは、展開したディレクトリーにあります。
3. **rhdm-7.0-controller-ee7** 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** としています。
4. **TEMP\_DIR/rhdm-7.0-controller-ee7/controller.war** ディレクトリーを **EAP\_HOME/standalone/deployments/** にコピーします。



### 警告

コピーするスタンドアロン Decision Server Controller デプロイメントの名前が Red Hat JBoss EAP 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。

5. **TEMP\_DIR/rhdm-7.0-controller-ee7/SecurityPolicy/** ディレクトリーの内容を **EAP\_HOME/bin** にコピーします。ファイルの上書きを確認されたら、**はい** を選択します。
6. **EAP\_HOME/standalone/deployments/** ディレクトリーに、**controller.war.dodeploy** という名前で空のファイルを作成します。このファイルにより、サーバーが起動するとスタンドアロン Decision Server Controller が自動的にデプロイされます。

#### 2.6.1.1. ユーザーの作成

スタンドアロン Decision Server Controller を使用する前に、**kie-server** ロールを持つユーザーを作成する必要があります。

#### 前提条件

**EAP\_HOME** ホームにインストールされたコントローラー

#### 手順

1. ターミナルアプリケーションで **EAP\_HOME/bin** 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、**<USER\_NAME>** および **<PASSWORD>** を、ご自分で考えたユーザー名およびパスワードに置き換えます。

```
$ ./add-user.sh -a --user <username> --password <password> --role kie-server
```



### 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を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。

3. ユーザー名とパスワードを書き留めておきます。

#### 2.6.1.2. Decision Server およびスタンドアロン Decision Server Controller の設定

Decision Server がスタンドアロン Decision Server Controller により管理される場合は、本セクションの説明に従って Decision Server およびスタンドアロン Decision Server Controller システム両方の **standalone.xml** ファイルを編集する必要があります。

## 前提条件

- 以下のセクションのいずれかに記載された手順に従って **EAP\_HOME** にインストールされた Decision Server
  - 「対話モードのインストーラーを使用した Red Hat Decision Manager のインストール」
  - 「CLI モードのインストーラーを使用した Red Hat Decision Manager のインストール」
  - 「ZIP ファイルを使用した Red Hat JBoss EAP への Red Hat Decision Manager のインストール」
- EAP\_HOME** にインストールされた Controller



### 注記

Red Hat では、実稼働環境では Decision Server とスタンドアロン Decision Server Controller を異なるサーバー上にインストールすることを推奨します。ただし、Decision Server とスタンドアロン Decision Server Controller を同じサーバー上にインストールする場合には (たとえば開発環境として)、共有する **standalone.xml** ファイルに対してこれらの変更を行います。

- Decision Server ノード: **kie-server** ロールを持つユーザー
- Controller サーバーノード: **kie-server** ロールを持つユーザー  
詳細については、「[ユーザーの作成](#)」を参照してください。

## 手順

- Controller の **EAP\_HOME/standalone/configuration/standalone.xml** ファイルにおいて、**<system-properties>** セクションに以下のプロパティを追加し、**<USERNAME>** および **<USER\_PWD>** を **kie-server** ロールを持つユーザーの認証情報に置き換えます。

```
<property name="org.kie.server.user" value="<USERNAME>"/>
<property name="org.kie.server.pwd" value="<USER_PWD>"/>
```

- Decision Server の **EAP\_HOME/standalone/configuration/standalone.xml** ファイルにおいて、**<system-properties>** セクションに以下のプロパティを追加します。

```
<property name="org.kie.server.controller.user" value="
<CONTROLLER_USER>"/>
<property name="org.kie.server.controller.pwd" value="
<CONTROLLER_PWD>"/>
<property name="org.kie.server.id" value="<KIE_SERVER_ID>"/>
<property name="org.kie.server.location" value="http://<HOST>:
<PORT>/kie-server/services/rest/server"/>
<property name="org.kie.server.controller" value="
<CONTROLLER_URL>"/>
```

- このファイルで、以下の値を置き換えます。
  - <CONTROLLER\_USER>** および **<CONTROLLER\_PWD>** を **kie-server** ロールを持つユーザーの認証情報に置き換える。



- **<KIE\_SERVER\_ID>** を Decision Server システムの ID または名前に置き換える (例: **rhdm700-decision-server-1**)。
- **<HOST>** を Decision Server ホストの ID または名前に置き換える (例: **localhost** または **192.7.8.9**)。
- **<PORT>** を Decision Server ホストのポートに置き換える (例: **8080**)。



#### 注記

**org.kie.server.location** プロパティで Decision Server の場所を指定します。

- **<CONTROLLER\_URL>** をスタンドアロン Decision Server Controller の URL に置き換える。Decision Server は、起動時にこの URL に接続されます。以下に例を示します。  
<http://<HOST>:<PORT>/controller/rest/controller>

### 2.6.1.3. スタンドアロン Decision Server Controller の実行

Red Hat JBoss EAP 上にスタンドアロン Decision Server Controller をインストールしたら、以下の手順を使用してスタンドアロン Decision Server Controller を実行します。

#### 前提条件

**EAP\_HOME** にインストールされ設定されたスタンドアロン Decision Server Controller

#### 手順

1. ターミナルアプリケーションで **EAP\_HOME/bin** に移動します。
2. 以下のコマンドを入力します。

- Linux または UNIX ベースのシステムの場合:

```
$ ./standalone.sh
```

- Windows の場合:

```
standalone.bat
```

3. Controller が Red Hat JBoss EAP 上で動作していることを確認するには、以下のコマンドを入力します。ここで、**<CONTROLLER>** と **<CONTROLLER\_PWD>** は、「[ユーザーの作成](#)」で作成したユーザー名とパスワードの組み合わせです。このコマンドにより、Decision Server インスタンスに関する情報が出力されます。

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



#### 注記

あるいは、Decision Server Java API Client を使用してスタンドアロン Decision Server Controller にアクセスすることもできます。

## 2.6.2. Red Hat JBoss Web Server へのスタンドアロン Decision Server Controller のインストール

スタンドアロン Decision Server Controller をインストールし、REST API または Decision Server Java Client API を使用して Controller を操作できます。

### 前提条件

- バックアップを作成してある Red Hat JBoss Web Server 3.1 with Tomcat 8 (またはそれ以降の) サーバーシステム。JBoss Web Server システムのベースディレクトリーは、**JWS\_HOME** と表示しています。
- インストールを完了するのに十分なユーザーパーミッション
- **rhdm-7.0-controller-jws.zip** ファイル。このファイルをダウンロードするには、以下の手順を実施します。
  - a. [Red Hat カスタマーポータル](#) の「Software Downloads」ページで、**Red Hat Decision Manager 7.0.0 Add-On's** の横の **Download** をクリックします。
  - b. **rhdm-7.0.0.GA-add-ons.zip** ファイルを展開します。**rhdm-7.0-controller-jws.zip** ファイルは、展開したディレクトリーにあります。

### 手順

1. **rhdm-7.0-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、このディレクトリーを **TEMP\_DIR** としています。
2. **TEMP\_DIR/rhdm-7.0-controller-jws.zip/controller.war** ディレクトリーを **JWS\_HOME/tomcat8/webapps** ディレクトリーにコピーします。



#### 警告

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。

3. **controller.war** フォルダーから **.war** 拡張子を削除します。
4. **TEMP\_DIR/rhdm-7.0-controller-jws/SecurityPolicy/** ディレクトリーの内容を **JWS\_HOME/bin** にコピーします。ファイルの上書きを確認されたら、**はい** を選択します。
5. **JWS\_HOME/standalone/deployments/** ディレクトリーに、**controller.war.dodeploy** という名前で空のファイルを作成します。このファイルにより、サーバーが起動するとスタンドアロン Decision Server Controller が自動的にデプロイされます。
6. **JWS\_HOME/tomcat8/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、**<USERNAME>** と **<PASSWORD>** はご自分で考えたユーザーとパスワードの組み合わせです。

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

- Decision Server を実行しているインスタンスの **JWS\_HOME/tomcat8/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。

```
CATALINA_OPTS="-Xmx1024m -Dorg.jboss.server.ext.disabled=true -
Dorg.jboss.ui.server.ext.disabled=true -
Dorg.jboss.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"
```

- スタンドアロン Decision Server Controller を実行しているインスタンスの **JWS\_HOME/tomcat8/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。

```
CATALINA_OPTS="-Dorg.kie.server.user=<USERNAME> -Dorg.kie.server.pwd=
<USER_PWD>"
```

- スタンドアロン Decision Server Controller を起動するには、**JWS\_HOME/tomcat8/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
```

- 数分後に **JWS\_HOME/tomcat8/logs** ディレクトリーを確認し、エラーが発生していたら修正します。

- Controller が Red Hat JBoss Web Server 上で動作していることを確認するには、以下のコマンドを入力します。このコマンドで、**<CONTROLLER>** および **<CONTROLLER\_PWD>** を **tomcat-users.xml** ファイル内の値に置き換えます。このコマンドにより、Decision Server インスタンスに関する情報が出力されます。

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



## 注記

あるいは、Decision Server Java API Client を使用してスタンドアロン Decision Server Controller にアクセスすることもできます。

## 第3章 関連ツールのインストール

### 3.1. プロジェクトでの **MAVEN** リポジトリの使用

Red Hat Decision Manager は、Red Hat Decision Manager Maven リポジトリと共に使用するように作られています。以下の方法のいずれかを用いて、ご自分のプロジェクトで Maven が Red Hat Decision Manager Maven リポジトリを使用するように設定できます。

- プロジェクトオブジェクトモデル (POM) ファイル (pom.xml) を設定する。
- Maven **settings.xml** ファイルを修正する。このファイルは、[Red Hat カスタマーポータル](#) からダウンロード可能な Maven リポジトリに含まれています。

Red Hat では、Maven グローバル設定またはユーザー設定を使用して、すべてのプロジェクトで Maven が Red Hat Decision Manager Maven リポジトリを使用するように設定することを推奨します。

#### 3.1.1. プロジェクト設定ファイル (pom.xml) を使用した **Maven** の設定

Maven を使用してご自分の Red Hat Decision Manager プロジェクトをビルドおよび管理するには、Maven を使用してプロジェクトがビルドされるように設定する必要があります。Maven には、プロジェクトの設定詳細を保持する POM ファイル (pom.xml) が必要です。詳細については、「[Apache Maven Project](#)」を参照してください。

##### 手順

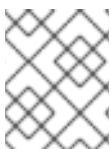
1. Maven プロジェクトを生成します。pom.xml ファイルは、Maven プロジェクトの作成時に自動的に生成されます。
2. pom.xml を編集して、追加の依存関係および新たなリポジトリを追加します。  
プロジェクトのコンパイルおよびパッケージング時に、Maven がすべての JAR ファイルおよび依存関係にある JAR ファイルを Maven リポジトリからダウンロードします。

pom.xml ファイルのスキーマは、[http://maven.apache.org/maven-v4\\_0\\_0.xsd](http://maven.apache.org/maven-v4_0_0.xsd) から取得できます。POM ファイルの詳細については、「[Apache Maven Project](#)」を参照してください。

#### 3.1.2. 設定ファイルを使用した **Maven** の設定

Maven 設定ファイル (settings.xml) を使用して、Maven の実行を設定します。このファイルは、以下の場所にあります。

- Maven インストールディレクトリー (\$M2\_HOME/conf/settings.xml)。これらの設定は、グローバル設定と呼ばれます。
- ユーザーのインストールディレクトリー (\$USER\_HOME/.m2/settings.xml)。これらの設定は、ユーザー設定と呼ばれます。
- システムプロパティー **kie.maven.settings.custom** で指定した特定の場所。



##### 注記

使用される設定は、これらの場所にあるファイルをマージしたものです。詳細については、「[Apache Maven Project](#)」を参照してください。

### 3.1.3. Maven 依存関係の管理

#### 前提条件

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** ファイルに該当する BOM (bill of materials) ファイルを追加する必要があります。BOM ファイルを追加すると、提供される Maven リポジトリから、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。

詳細については、「[What is the mapping between RHDM product and maven library version?](#)」を参照してください。

#### 手順

1. **pom.xml** で BOM を宣言します。以下に例を示します。

##### 例3.1 Red Hat Decision Manager 7.0.0 用の BOM

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.jboss.bom.rhdm</groupId>
      <artifactId>rhdm-platform-bom</artifactId>
      <version>7.0.0.Final-redhat-4</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
<dependencies>
  <!-- Your dependencies -->
</dependencies>
```

2. プロジェクトに必要な依存関係を **<dependencies>** タグで宣言します。

- 標準的な Red Hat ビジネス自動化プロジェクトでは、以下の依存関係を宣言します。

##### 埋め込まれた jBPM エンジン依存関係

```
<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>

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

```

- CDI を使用する Red Hat ビジネス自動化プロジェクトでは、以下の依存関係を宣言します。

### CDI 対応 jBPM エンジン依存関係

```

<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>

```

- 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。

### 埋め込まれた Drools 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:drools-workbench-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>
<dependency>
  <groupId>org.kie.server</groupId>
  <artifactId>kie-server-api</artifactId>
</dependency>

<!-- Dependency for Red Hat JBoss BRMS functionality. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-core</artifactId>
</dependency>

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

```

- Red Hat ビジネス自動化または Red Hat Decision Manager の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。

#### クライアント依存関係

```

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

```

- KJAR パッケージのアセットを使用するには、**kie-maven-plugin** を含める方法を推奨します。

#### Kie Maven プラグイン

```

<packaging>kjar</packaging>
<build>
  <plugins>
    <plugin>

```

```

<groupId>org.kie</groupId>
<artifactId>kie-maven-plugin</artifactId>
<version>7.0.0.Final-redhat-4</version>
<extensions>true</extensions>
</plugin>
</plugins>
</build>

```

## 3.2. GIT リポジトリ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あり、リビジョンをコミットオブジェクトとして実装します。変更をリポジトリにコミットすると、Git リポジトリに新規コミットオブジェクトが作成されます。Decision Central でプロジェクトを作成すると、そのプロジェクトが Decision Central に接続された Git リポジトリに追加されます。

他の Git リポジトリにプロジェクトがある場合は、それらを Decision Central のスペースにインポートすることができます。

### 前提条件

外部の Git リポジトリにある Red Hat Decision Manager プロジェクト

### 手順

1. Decision 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 **myteam** 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** ウィンドウにおいて、インポートするプロジェクトが含まれる Git リポジトリの URL および認証情報を入力し、**Import** をクリックします。プロジェクトが現在のスペースに追加されます。

## 3.3. RED HAT JBOSS DEVELOPER STUDIO のインストールおよび設定

Red Hat JBoss Developer Studio は Eclipse をベースにした JBoss 統合開発環境 (IDE) です。Red Hat JBoss Developer Studio は Red Hat Decision Manager 用のツールおよびインターフェースを持つプラグインを提供します。

### 手順

1. [Red Hat カスタマーポータル](#) から最新の Red Hat JBoss Developer Studio をダウンロードします。
2. **Red Hat JBoss Developer Studio** のドキュメントに記載された設定およびインストール手順に従います。
3. 次のセクションに記載される手順に従って、Red Hat JBoss Developer Studio プラグインをインストールします。





## 重要

複数バイトのルール名を処理する方法に関する問題により、JBoss Developer Studio のインスタンスは、ファイルエンコーディングを **UTF-8** に設定して起動する必要があります。そのためには、**\$JBDS\_HOME/studio/devstudio.ini** ファイルを編集し、**"-Dfile.encoding=UTF-8"** というプロパティを追加します。

### 3.3.1. Red Hat JBoss Developer Studio プラグインのインストール

Red Hat JBoss Developer Studio は Red Hat Decision Manager 用のツールおよびインターフェースを持つプラグインを提供します。これらのプラグインはコミュニティバージョンの製品が基になっています。そのため、Red Hat Decision Manager プラグインは Drools プラグインと呼ばれます。

[Red Hat カスタマーポータル](#) から最新の Red Hat JBoss Developer Studio を取得します。Red Hat JBoss Developer Studio の Red Hat Decision Manager プラグインは、更新サイトより入手できます。

#### 手順

1. Red Hat JBoss Developer Studio を起動します。
2. **Help** → **Install New Software** をクリックします。
3. **Add** をクリックして **Add Repository** メニューに入ります。
4. **Name** フィールドの横で名前を指定し、**Location** フィールドに <https://devstudio.jboss.com/11/stable/updates/integration-stack/> の URL を追加します。
5. **OK** をクリックします。
6. 利用可能なオプションから **JBoss Business Process and Rule Development** 機能を選択して **Next** をクリックし、もう一度 **Next** をクリックします。
7. ライセンス内容を確認して、適切なラジオボタンを押してライセンス内容に同意し、**Finish** をクリックします。
8. インストールプロセスが終了したら、Red Hat JBoss Developer Studio を再起動します。

### 3.3.2. Red Hat Decision Manager サーバーの設定

Red Hat Decision Manager サーバーを実行するように Red Hat JBoss Developer Studio を設定することができます。

#### 前提条件

- Red Hat JBoss Developer Studio
- Red Hat JBoss Developer Studio プラグインと共にインストールした Red Hat Decision Manager

#### 手順

1. Red Hat JBoss Developer Studio を起動します。

2. Drools ビューを開くには、**Window** → **Open Perspective** → **Other** の順にクリックし、**Drools** を選択し、特定のビューを選択して **OK** をクリックします。
3. **Window** → **Show View** → **Other...** の順にクリックし、**Server** → **Servers** を選択してサーバービューを追加します。
4. **Servers** パネルを右クリックし、**New** → **Server** を選択してサーバーメニューを開きます。
5. **JBoss Enterprise Middleware** → **JBoss Enterprise Application Platform 7.1+** をクリックし、**Next** をクリックしてサーバーを定義します。
6. **Browse** ボタンをクリックしてホームディレクトリーを設定します。Red Hat Decision Manager をインストールした Red Hat JBoss EAP ディレクトリーに移動します。  
Red Hat Decision Manager サーバーを設定するには、Red Hat Decision Manager をインストールした Red Hat JBoss EAP ディレクトリーを選択します。
7. **Name** フィールドにサーバーの名前を指定して、設定ファイルが設定されていることを確認し、**Finish** をクリックします。

### 3.3.3. Git リポジトリから Red Hat JBoss Developer Studio へのプロジェクトのインポート

Red Hat JBoss Developer Studio は、集中 Git アセットリポジトリに接続するように設定できます。リポジトリにはルール、モデル、関数、およびプロセスが保存されます。

ローカルの Git リポジトリをインポートする、またはリモート Git リポジトリのクローンを作成することができます。

### 3.3.4. リモート Git リポジトリのクローン作成

Git リポジトリのクローンを作成して Red Hat JBoss Developer Studio で使用できます。

#### 前提条件

クローンを作成するリモート Git リポジトリへのアクセスパーミッション

#### 手順

1. Red Hat JBoss Developer Studio の **Server** タブからサーバーを選択し、Start アイコンをクリックして選択したサーバーを起動します。
2. セキュアシェルサーバーがまだ実行されていなければ、ターミナルに以下のコマンドを入力してサーバーを起動します。

```
/sbin/service sshd start
```



#### 注記

このコマンドは、Linux および Apple Macintosh 固有のもので、これらのプラットフォームで **sshd** がすでに起動していると、このコマンドに失敗しますが、その場合はこのステップを無視しても問題ありません。

3. Red Hat JBoss Developer Studio で **File** → **Import...** を選択し、Git フォルダーに移動します。Git フォルダーを開き、**Projects from Git** を選択して **Next** をクリックします。

4. リポジトリのソースに **Clone URI** を選択し、**Next** をクリックします。
5. 次のウィンドウで Git リポジトリの詳細を入力して、**次へ** をクリックします。
6. 次のウィンドウでインポートするブランチを選択し、**Next** をクリックします。
7. このプロジェクトのローカルストレージを定義するために、空ではないディレクトリを入力 (または選択) して設定を変更し、**Next** をクリックします。
8. 次のウィンドウでプロジェクトを一般プロジェクトとしてインポートし、**Next** をクリックします。
9. このプロジェクトに名前を付け、**Finish** をクリックします。

### 3.3.5. ローカル Git リポジトリのインポート

ローカルの Git リポジトリをインポートして Red Hat JBoss Developer Studio で使用することができます。

#### 手順

1. **Server** タブからサーバーを選択し、Start アイコンをクリックしてサーバーを起動します。
2. Red Hat JBoss Developer Studio で **File** → **Import...** を選択し、Git フォルダーに移動します。Git フォルダーを開き、**Projects from Git** を選択して **Next** をクリックします。
3. リポジトリのソースに **Existing local repository** を選択し、**Next** をクリックします。
4. 使用可能なリポジトリの一覧から設定するリポジトリを選択し、**Next** をクリックします。
5. 表示されたダイアログウィンドウで、**Wizard for project import** グループから **Import as general project** ラジオボタンを選択し、**Next** をクリックします。
6. このプロジェクトに名前を付け、**Finish** をクリックします。

## 第4章 RED HAT DECISION MANAGER におけるクラスター化 (デザインタイム開発環境向け)



### 注記

本セクションは、Git リポジトリをクラスター化する必要がある Decision Central の開発環境に主眼を置いています。Decision Server の実稼働環境では、クラスター環境を構築する必要はありません。

Red Hat では、デザインタイム開発環境向けに、2 つまたはそれ以上のコンピューターをクラスター化した高可用性クラスター環境の構築について検討することを推奨します。これにより、協調作業が促進され高可用性が得られます。

たとえば、クラスター化が設定されている場合、開発者 X が Decision Central ノード 1 でルールを作成すると、Decision Central ノード 2 で作業中の開発者 Y がそのルールを見ることができます。さらに、クラスター環境の各ノードで作業中の開発者全員が、同一コンテンツを同期された状態で見ることができます。

また、クラスター化によりルールの開発環境に高可用性がもたらされます。開発者 X がノード 1 で作業中にそのノードに異常が発生したとしても、開発者 X の作業内容が維持されクラスター環境のどのノードからでもその内容にアクセスすることができます。

以下のセクションでは、クラスター化された Red Hat Decision Manager 環境を構築する方法について説明します。

### 4.1. ELASTICSEARCH の設定

Elasticsearch は、スケーラビリティに優れたオープンソースの全文検索および解析エンジンです。これにより、大量のデータを素早くほぼリアルタイムに保存、検索、および解析できます。Red Hat Decision Manager のクラスター環境では、ノード全体にわたる複雑な検索を効率的に実施できます。Elasticsearch は、クラスター環境のメインノードに設定します。

#### 手順

1. [Elasticsearch](#) のインストールファイルをダウンロードして展開します。
2. `elasticsearch/config/elasticsearch.yml` ファイルの内容を以下のように置き換えます。ここで、`<MAIN_NODE_IP>` はクラスターのメインノードの IP アドレスです。

```
cluster.name: kie-cluster
transport.host: <MAIN_NODE_IP>
http.host: <MAIN_NODE_IP>
transport.tcp.port: 9300
xpack.security.enabled: false
discovery.zen.minimum_master_nodes: 1
```

3. 以下のプラグインをインストールします。

- Linux または UNIX ベースのシステムの場合は、以下のように入力します。

```
./bin/elasticsearch-plugin install x-pack
```

- Windows の場合は、以下のように入力します。

```
bin\elasticsearch-plugin.bat install x-pack
```

4. メインノードで Elasticsearch を実行します。

- Linux または UNIX ベースのシステムの場合は、以下のように入力します。

```
./bin/elasticsearch
```

Windows の場合は、以下のように入力します。

```
bin\elasticsearch.bat
```

## 4.2. ACTIVEMQ JMS ブローカーのアクティブ化

Elasticsearch をインストールしたら、クラスターのメインノードで ActiveMQ JMS ブローカーをアクティブ化する必要があります。

### 前提条件

クラスターのメインノードにインストールされた Elasticsearch

### 手順

1. **EAP\_HOME/standalone/configuration/standalone.xml** ファイルに以下のプロパティを追加します。
  - `socket-binding-group` 要素に `<socket-binding name="activemq" port="61616"/>` を追加します。
  - `messaging-activemq` サブシステム要素の `<server name="default">` に、`<remote-acceptor name="activemq-acceptor" socket-binding="activemq"/>` を追加します。
2. クラスターの各ノードについて、[「クラスターノードへの Decision Central のインストール」](#)に記載のステップをすべて実施します。

## 4.3. クラスターノードへの DECISION CENTRAL のインストール

本セクションのステップをすべて実施して、クラスターの各ノードに Decision Central をインストールして実行します。

### 前提条件

- すべての Red Hat Decision Manager ノードがアクセスできる NFS マウントしたディレクトリが利用可能
- クラスターのメインノードにインストールされた Elasticsearch
- クラスターのメインノードでアクティブ化した ActiveMQ JMS ブローカー

### 手順

1. クラスターの各ノード上に Decision Central をインストールします。Decision Central のインストールの詳細については、「[2章 Red Hat Decision Manager のインストール](#)」を参照してください。
2. `./standalone.xml` ファイルの以下のプロパティを編集します。ここで、
  - `<MAIN_NODE>` はクラスターのメインノードの IP アドレスです
  - `<JMS_BROKER_USER>` は JMS ブローカー用のユーザー名です
  - `<JMS_BROKER_PASSWORD>` は JMS ブローカー用のパスワードです

```
<system-properties>
  <property name="org.uberfire.nio.git.dir" value="
    <niogit_dir_on_shared_nfs>"/>
  <property name="appformer-cluster" value="true"/>
  <property name="appformer-jms-url"
value="tcp://<MAIN_NODE_IP>:61616"/>
  <property name="appformer-jms-username" value="
<JMS_BROKER_USER>"/>
  <property name="appformer-jms-password" value="
<JMS_BROKER_PASSWORD>"/>
  <property name="org.appformer.ext.metadata.index"
value="elastic"/>
  <property name="org.appformer.ext.metadata.elastic.port"
value="9300"/>
  <property name="org.appformer.ext.metadata.elastic.host"
value="<MAIN_NODE_IP>"/>
  <property name="org.appformer.ext.metadata.elastic.cluster"
value="kie-cluster"/>
  <property name="org.appformer.ext.metadata.elastic.retries"
value="10"/>
</system-properties>
```

3. クラスター環境のメインノードにインストールした Red Hat JBoss EAP において、**admin** ロールを持つユーザーを作成します。この時、ユーザー名およびパスワードは、前のステップで作成した `<JMS_BROKER_USER>` および `<JMS_BROKER_PASSWORD>` の値と一致させます。

```
$ <MAIN_NODE_EAP_HOME>/bin/. /add-user.sh -a --user <JMS_BROKER_USER>
--password <JMS_BROKER_PASSWORD> --role admin
```

4. Decision Central を起動するには、クラスターの各ノードにお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:

```
EAP_HOME/bin/standalone.sh
```

- Windows の場合:

```
EAP_HOME\bin\standalone.bat
```

## 第5章 RED HAT DECISION MANAGER システムの検証

Red Hat Decision Manager をインストールしたら、アセットを作成してシステムが機能していることを検証します。

### 手順

1. 以下のコマンドを入力して Decision Server を起動します。

```
EAP_HOME/bin/standalone.sh
```

2. Web ブラウザーで **localhost:8080/decision-central** と入力します。Red Hat Decision Manager をドメイン名から実行するように設定している場合は、以下のように **localhost** をドメイン名に置き換えます。

<http://www.example.com:8080/decision-central>

Red Hat Decision Manager をクラスター環境で実行するように設定している場合は、以下のよう  
に **localhost** を特定ノードの IP アドレスに置き換えます。

[http://<node\\_IP\\_address>:8080/decision-central](http://<node_IP_address>:8080/decision-central)

3. インストール時に作成した管理者ユーザーの認証情報を入力します。Decision Central ホーム  
ページが表示されます。
4. **Menu** → **Design** → **Projects** の順に選択します。
5. **Try Samples** をクリックします。
6. **mortgages** をクリックします。Assets ウィンドウが表示されます。
7. **Create New Asset** をクリックします。
8. **Data Object** を選択します。
9. **Name** フィールドに MyDataObject と入力し、**OK** をクリックします。
10. **Spaces** → **myteam** → **mortgages** の順にクリックし、MyDataObject がアセットリストにある  
ことを確認します。
11. クラスター化したシステムを検証する場合は、以下の URL を入力します。ここ  
で、<node\_IP\_address> は別のノードのアドレスです。MyDataObject アセットを作成した  
Decision Central で使用したものと同一認証情報を入力します。  
[http://<node\\_IP\\_address>:8080/decision-central](http://<node_IP_address>:8080/decision-central)
12. **Menu** → **Design** → **Projects** の順に選択します。
13. **mortgages** プロジェクトを選択します。
14. MyDataObject がアセットリストにあることを確認します。
15. MyDataObject を削除します。

## 第6章 DECISION CENTRAL のカスタマイズ

### 6.1. DECISION CENTRAL ログインページのカスタマイズ

個別のビジネスニーズに合わせて、Decision Central のログインページをカスタマイズできます。これには、企業のロゴやプロジェクトロゴが含まれます。

#### 手順

1. Red Hat JBoss EAP を起動し、Web ブラウザーで Decision Central を開きます。
2. Red Hat Decision Manager システムの **EAP\_HOME/standalone/deployments/decision-central.war/img/** ディレクトリーに移動します。
3. ログインページの右上隅に表示される企業のロゴを変更するには、デフォルト画像 **login-screen-logo.png** を PNG 形式の新しい画像に置き換えます。
4. **User name** および **Password** フィールドの上に表示されるプロジェクトロゴを変更するには、デフォルト画像 **RHDM\_Logo.svg** を新しい SVG ファイルに置き換えます。
5. ログインページすべてを強制的に再読み込みし、キャッシュを消去して変更を表示します。たとえば、Linux および Windows の多くの Web ブラウザーでは、Ctrl+F5 を押します。

### 6.2. DECISION CENTRAL アプリケーションヘッダーのカスタマイズ

個別のビジネスニーズに合わせて、Decision Central のアプリケーションヘッダーをカスタマイズすることができます。

1. Red Hat JBoss EAP を起動し、Web ブラウザーで Decision Central を開き、ご自分のユーザー認証情報でログインします。
2. SVG 形式の新しいアプリケーションヘッダー画像を、Red Hat Decision Manager システムの **EAP\_HOME/standalone/deployments/decision-central.war/banner/** ディレクトリーにコピーします。
3. テキストエディターで **EAP\_HOME/standalone/deployments/decision-central.war/banner/banner.html** ファイルを開きます。
4. **banner.html** ファイルで以下の **<img>** タグを編集し、新しいヘッダー画像の名前を指定します。

```

```

5. ログインページすべてを強制的に再読み込みし、キャッシュを消去して変更を表示します。たとえば、Linux および Windows の多くの Web ブラウザーでは、Ctrl+F5 を押します。



## 第7章 RED HAT DECISION MANAGER と RED HAT SINGLE SIGN-ON の統合

Red Hat Single Sign-On (RH-SSO) はシングルサインオンソリューションで、これを使用してブラウザーアプリケーションと REST Web サービス、および Git へのアクセスのセキュリティを確保できます。本章では、RH-SSO と Red Hat Decision Manager を統合してこの機能を活用する方法を説明します。

RH-SSO と統合すると、Red Hat Decision Manager 向けに SSO と IDM (アイデンティティ管理) の統合環境を利用できます。RH-SSO のセッション管理機能により、1 度認証するだけで、Web 上でさまざまな Red Hat Decision Manager 環境を使用できます。

RH-SSO の詳しい情報は [RH-SSO のドキュメント](#) を参照してください。

### RH-SSO 統合ポイント

以下の統合ポイントを使用して Decision Server と RH-SSO を統合できます。

- **RH-SSO サーバーを使用した Red Hat Decision Manager の認証**  
RH-SSO を使用して Red Hat Decision Manager を認証するには、Red Hat Decision Manager の Web クライアントとリモートサービスの両方を RH-SSO によりセキュア化する必要があります。この統合により、RH-SSO を使用して Web インターフェースまたはリモートサービスコンシューマーのいずれかで Red Hat Decision Manager に接続することができます。
- **RH-SSO サーバーを使用した Decision Server の認証**  
サーバー認証に Web インターフェースが提供されないので、RH-SSO を使用して Red Hat Decision Manager の Decision Server を認証するには、Decision Server が提供するリモートサービスをセキュア化する必要があります。これにより、任意の Red Hat Decision Manager リモートサービスコンシューマー (ユーザーまたはサービス) が RH-SSO を使用して認証できます。
- **RH-SSO サーバーを使用したサードパーティークライアントの認証**  
RH-SSO サーバーを使用してサードパーティークライアントを認証するには、サードパーティークライアントが RH-SSO を使用して自身を認証し、Red Hat Decision Manager および Decision Server が提供する REST API またはリモートファイルシステムサービスなどのリモートサービスエンドポイントを消費する必要があります。

以下のセクションでは、これらの統合ポイントを使用して RH-SSO 統合を行う方法について説明します。

### 7.1. RH-SSO を使用した RED HAT DECISION MANAGER の認証

RH-SSO を使用して Red Hat Decision Manager を認証するには、以下の手順を実施します。

1. RH-SSO サーバーに Red Hat Decision Manager のレルムクライアントを設定して実行する。
2. Red Hat JBoss EAP の RH-SSO クライアントアダプターをインストールして設定する。
3. RH-SSO を使用して Red Hat Decision Manager リモートサービスのセキュリティを確保する。

#### 7.1.1. RH-SSO への Red Hat Decision Manager レルムクライアントの設定

セキュリティレルムは、異なるアプリケーションリソースのアクセスを制限するのに使用します。RH-SSO インスタンスが非公開か他の製品と共有されているかにかかわらず、新規レルムを作成する必要があります。マスターレルムを、スーパー管理者がシステムのレルムを作成して管理する場所として

維持できます。他の製品システムと共有されている RH-SSO インスタンスと統合して、これらのアプリケーションとのシングルサインオンを行うためには、これらのアプリケーションはすべて同じレルムを使用する必要があります。

## 手順

1. Red Hat カスタマーポータルの「[Software Downloads](#)」セクションから RH-SSO をダウンロードします。
2. 基本的な RH-SSO のスタンドアロンサーバーをインストールして設定します。インストールおよび設定は、『[Red Hat Single Sign On Getting Started Guide](#)』の「[Install and Boot](#)」の章に記載された手順に従ってください。実稼働環境の設定については、[Red Hat Single Sign On](#) の『[Server Administration Guide](#)』を参照してください。

### 注記

同じシステム上で RH-SSO と Red Hat Decision Manager サーバーの両方を実行する場合には、以下の手段のいずれかによりポートの競合を避けてください。

- 以下のよう  
に、`RHSSO_HOME/standalone/configuration/standalone.xml` ファイルを更新して、ポートのオフセットを 100 に設定する。

```
<socket-binding-group name="standard-sockets" default-interface="public" port-offset="${jboss.socket.binding.port-offset:100}">
```

- 環境変数を使用してサーバーを実行する。

```
bin/standalone.sh -Djboss.socket.binding.port-offset=100
```

3. RH-SSO サーバーを起動し、`RHSSO_HOME/bin` から RH-SSO を設定します。

```
./standalone.sh
```

RH-SSO サーバーが起動したら、Web ブラウザーで <http://localhost:8180/auth/admin> にアクセスし、RH-SSO のインストール時に作成した管理者の認証情報を使用してログインします。初回のログイン時に、新規ユーザー登録フォームで初期ユーザーを設定することができます。

4. RH-SSO 管理コンソールで、**Realm Settings** メニューアイテムをクリックします。
5. **Realm Settings** ページで **Add Realm** をクリックします。  
**Add realm** ページが表示されます。
6. **Add realm** ページで、レルムの名前を指定して **Create** をクリックします。
7. **Clients** メニューアイテムをクリックし、**Create** をクリックします。  
**Add Client** ページが表示されます。
8. **Add Client** ページで、以下のようにレルム用の新規クライアント作成に必要な情報を指定します。
  - **Client ID**: kie

- **Client protocol**: openid-connect
- **Root URL**: http://localhost:8080/decision-central

9. **Save** をクリックして変更を保存します。

作成した新規クライアントの **Access Type** は、デフォルトでは **public** に設定されています。この設定を **confidential** に変更します。

この時点で、RH-SSO サーバーに Red Hat Decision Manager アプリケーションのレلمおよびクライアントが設定され、**localhost:8180** で HTTP 接続をリッスンした状態で実行されます。このレلمは Red Hat Decision Manager アプリケーションに異なるユーザー、ロール、セッションを提供します。

### 7.1.2. Red Hat JBoss EAP の RH-SSO クライアントアダプターの設定

Red Hat JBoss EAP の RH-SSO クライアントアダプターを設定するには、Red Hat JBoss EAP の RH-SSO アダプターをインストールし、続いて Red Hat Decision Manager アプリケーションおよび RH-SSO クライアントアダプターを設定します。

#### 手順

1. Red Hat JBoss EAP 6.4.X または 7.0 をインストールします。  
バージョン 6 の場合は、『**Red Hat JBoss Enterprise Application Platform インストールガイド**』の「[インストールの説明](#)」の章を参照してください。  
  
バージョン 7 の場合は、『**Red Hat JBoss Enterprise Application Platform インストールガイド**』の「[JBoss EAP のインストール](#)」の章を参照してください。
2. 新規にインストールした Red Hat JBoss EAP ホームに Red Hat Decision Manager をインストールします。  
**standalone.xml** ファイルに変更を加えて RH-SSO アダプターを設定してから Red Hat Decision Manager を展開すると、RH-SSO アダプターの設定が上書きされて失われる可能性があります。
3. [Red Hat カスタマーポータル](#) から Red Hat JBoss EAP アダプターをダウンロードします。
4. アダプターを展開してインストールします。インストールの説明は、『**Red Hat Single Sign On Securing Applications and Services Guide**』の「[JBoss EAP Adapter](#)」セクションを参照してください。
5. バージョン 7 の場合は、**EAP\_HOME/standalone/configuration** に移動して **standalone.xml** および **standalone-full.xml** ファイルを開きます。各ファイルから **<single-sign-on/>** 要素を削除します。  
Red Hat JBoss EAP 6 の場合は、このステップを実行する必要はありません。

#### 手順

1. Red Hat JBoss EAP システムの **EAP\_HOME/standalone/configuration** ディレクトリーに移動し、**standalone.xml** ファイルを編集して RH-SSO サブシステム設定を追加します。以下に例を示します。

```
<subsystem xmlns="urn:jboss:domain:keycloak:1.1">
  <secure-deployment name="decision-central.war">
    <realm>demo</realm>
    <realm-public-
```

```
key>MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCrVrCuTtArbgaZzL1hvh0xtL5m
c7o0NqPVnYXkLvgcwIC3BjLGW1tGEGoJaXDuSaRllobm53JBhjx33UNv+5z/UMG4kytB
WxheNVKnL6GgqINabMaFfPLPCF8kAgKnsi79NMo+n6KnSY8YeUmec/p2vj02NjsSAVcW
EQMVhJ31LwIDAQAB</realm-public-key>
  <auth-server-url>http://localhost:8180/auth</auth-server-url>
  <ssl-required>external</ssl-required>
  <enable-basic-auth>true</enable-basic-auth>
  <resource>kie</resource>
  <credential name="secret">759514d0-dbb1-46ba-b7e7-
ff76e63c6891</credential>
  <principal-attribute>preferred_username</principal-attribute>
</secure-deployment>
</subsystem>
```

この例で、

- **secure-deployment name** は、アプリケーションの WAR ファイルの名前です。
- **realm** は、使用するアプリケーション用に作成したレルムの名前です。
- **realm-public-key** は、作成したレルムの公開鍵です。この鍵は、RH-SSO 管理コンソールで作成したレルムの **Realm settings** ページの **Keys** タブで確認することができます。**realm-public-key** の値を指定しない場合は、サーバーが自動的に取得します。
- **auth-server-url** は、RH-SSO 認証サーバーの URL です。
- **enable-basic-auth** は、クライアントがトークンベースと基本認証アプローチの両方を使用して要求を実行できるように、基本認証メカニズムを有効にする設定です。
- **resource** は、作成したクライアントの名前です。
- **credential name** は、作成したクライアントの秘密鍵です。この鍵は、RH-SSO 管理コンソールの **Clients** ページの **Credentials** タブで確認することができます。
- **principal-attribute** は、ユーザーのログイン名です。この値を指定しないと、ユーザー名ではなくユーザー ID がアプリケーションに表示されます。



### 注記

RH-SSO サーバーは、ユーザー名を小文字に変換します。そのため、RH-SSO と統合した後は、Red Hat Decision Manager ではユーザー名が小文字で表示されます。ビジネスプロセスでユーザー名が大文字でハードコード化されている場合には、アプリケーションが大文字のユーザー名を識別できない場合があります。

2. **EAP\_HOME/bin/** に移動し、以下のコマンドを入力して Red Hat JBoss EAP サーバーを起動します。

```
./standalone.sh
```



## 注記

RH-SSO セキュリティーサブシステムを使用するようにアプリケーションの WAR ファイルを更新して、Red Hat JBoss EAP の RH-SSO アダプターを設定することもできます。ただし Red Hat では、RH-SSO サブシステムからアダプターを設定することを推奨します。つまり、設定を各 WAR ファイルに適用するのではなく、Red Hat JBoss EAP の設定を更新します。

### 7.1.3. 新規ユーザーの追加

新規ユーザーを追加して Red Hat Decision Manager にアクセスするためのロールを割り当てるには、以下の手順を実施します。

1. RH-SSO 管理コンソールにログインして、ユーザーを追加するレلمを開きます。
2. **Manage** セクションで **Users** メニューアイテムをクリックします。  
**Users** という名前の空のユーザー一覧のページが表示されます。
3. 空のユーザー一覧で **Add User** ボタンをクリックして、新規ユーザーの作成を開始します。  
**Add user** ページが表示されます。
4. **Add user** ページでユーザー情報を指定して、**Save** をクリックします。
5. **Credentials** タブで新規パスワードを設定します。
6. **admin** または **analyst** ロールなど、Red Hat Decision Manager へのアクセスが許可されるロールの 1 つを新規ユーザーに割り当てます。  
**Roles** セクションの **Realm Roles** タブで、このロールをレلمロールとして定義します。
7. **Users** ページの **Role Mappings** タブをクリックしてロールを割り当てます。

これで、Decision Server を起動した後に Decision Central にログインできるようになります。

### 7.1.4. RH-SSO を使用した Red Hat Decision Manager リモートサービスのセキュリティー確保

Red Hat Decision Manager は、リモート API を使用してサードパーティークライアントが消費可能なリモートサービスエンドポイントを各種提供します。RH-SSO でこれらのサービスを認証するには、**BasicAuthSecurityFilter** と呼ばれるセキュリティーフィルターを無効化する必要があります。

#### 手順

1. アプリケーションデプロイメント記述子ファイル (**WEB-INF/web.xml**) を開き、以下の変更を加えます。
  - 以下の行を削除して、**org.uberfire.ext.security.server.BasicAuthSecurityFilter** クラスのサーブレットフィルターとマッピングを削除します。

```
<filter>
  <filter-name>HTTP Basic Auth Filter</filter-name>
  <filter-
class>org.uberfire.ext.security.server.BasicAuthSecurityFilter</f
ilter-class>
```

```

    <init-param>
      <param-name>realmName</param-name>
      <param-value>KIE Workbench Realm</param-value>
    </init-param>
  </filter>

  <filter-mapping>
    <filter-name>HTTP Basic Auth Filter</filter-name>
    <url-pattern>/rest/*</url-pattern>
    <url-pattern>/maven2/*</url-pattern>
    <url-pattern>/ws/*</url-pattern>
  </filter-mapping>

```

- 以下の行を追加して、フィルターマッピングから削除した url-patterns の **security-constraint** を追加します。

```

<security-constraint>
  <web-resource-collection>
    <web-resource-name>remote-services</web-resource-name>
    <url-pattern>/rest/*</url-pattern>
    <url-pattern>/maven2/*</url-pattern>
    <url-pattern>/ws/*</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <role-name>rest-all</role-name>
  </auth-constraint>
</security-constraint>

```

2. 変更を保存します。

### 7.1.5. RH-SSO を使用した Red Hat Decision Manager ファイルシステムサービスのセキュリティ確保

ファイルシステムなど、他のリモートサービス (例: リモート GIT サービス) を消費するには、正しい RH-SSO ログインモジュールを指定する必要があります。最初に、JSON 設定ファイルを生成します。

#### 手順

1. <http://localhost:8080/auth/admin> から **RH-SSO 管理コンソール** に移動します。
2. **Clients** をクリックします。
3. 以下の設定で新規クライアントを作成します。
  - **Client ID** は **kie-git** に設定します。
  - **Access Type** は **confidential** に設定します。
  - **Standard Flow Enabled** オプションを無効化します。
  - **Direct Access Grants Enabled** オプションを有効化します。



Clients &gt; kie-git

Kie-git Settings

Credentials

Roles

Mappers Scope 

Revocation

Sessions Offline Access 

Clustering

Installation Client ID 

kie-git

Name Description Enabled ☒ ONConsent Required ☐ OFFClient Protocol 

openid-connect

Client Template Access Type 

confidential

Standard Flow Enabled ☐ OFFDirect Access Grants Enabled ☒ ONService Accounts Enabled ☐ OFFRoot URL Base URL Admin URL 

Save

Cancel

4. **Save** をクリックします。

5. クライアント設定画面上部にある **Installation** タブをクリックして、**Format Option** に **Keycloak OIDC JSON** を選択します。

6. **Download** をクリックします。

7. ダウンロードした JSON ファイルを、サーバーのファイルシステム内でアクセス可能なディレクトリに移動するか、アプリケーションクラスパスに追加します。

詳しい情報は、Keycloak の『**Securing Applications and Services Guide**』の「[JAAS plugin](#)」の章を参照してください。

JSON 設定ファイルを正常に生成してダウンロードしたら、正しい RH-SSO ログインモジュールを **EAP\_HOME/standalone/configuration/standalone.xml** ファイルで指定します。デフォルトでは、Red Hat Decision Manager のセキュリティードメインは **other** に設定されています。このセキュリティードメインの **login-module** のデフォルト値を、以下の例で示す値に置き換えます。

```
<security-domain name="other" cache-type="default">
  <authentication>
    <login-module
code="org.keycloak.adapters.jaas.DirectAccessGrantsLoginModule"
flag="required">
      <module-option name="keycloak-config-file" value="$EAP_HOME/kie-
git.json"/>
    </login-module>
  </authentication>
</security-domain>
```

**module-option** 要素で指定した JSON ファイルには、リモートサービスのセキュリティーを確保するために使用するクライアントが含まれます。**module-option** 要素の **\$EAP\_HOME/kie-git.json** の

値を、この JSON 設定ファイルの絶対パスまたはクラスパス (**classpath:/EXAMPLE\_PATH/kie-git.json**) に置き換えます。

この時点で RH-SSO サーバーで認証されたすべてのユーザーは、内部 GIT リポジトリのクローンを作成することができます。以下のコマンドの **USER\_NAME** を、**admin** などの RH-SSO ユーザーに変更します。

```
git clone ssh://USER_NAME@localhost:8001/system
```

### 7.1.6. RH-SSO のユーザーおよびグループ管理の有効化

本セクションでは、Decision Central を使用して RH-SSO に保存されたユーザーおよびグループを管理する方法について説明します。

#### 手順

1. 以下のライブラリーが **WEB-INF/lib** ディレクトリーにあることを確認します。

```
uberfire-security-management-api-<latest_artifact_version>.jar
uberfire-security-management-backend-<latest_artifact_version>.jar
uberfire-security-management-keycloak-<latest_artifact_version>.jar
keycloak-core-<latest_artifact_version>.jar
keycloak-common-<latest_artifact_version>.jar
```

2. サードパーティーセキュリティ JAR ファイルを削除します。以下に例を示します。

```
uberfire-security-management-wildfly-<latest_artifact_version>.jar
uberfire-security-management-tomcat-<latest_artifact_version>.jar
```

3. **WEB-INF/classes/security-management.properties** ファイルの内容全体を、以下の内容に置き換えます。

```
org.uberfire.ext.security.management.api.userManagementServices=KCCredentialsUserManagementService
org.uberfire.ext.security.management.keycloak.authServer=http://localhost:8081/auth
org.uberfire.ext.security.management.keycloak.realm=demo
org.uberfire.ext.security.management.keycloak.user=admin
org.uberfire.ext.security.management.keycloak.password=admin
org.uberfire.ext.security.management.keycloak.clientId=kie
org.uberfire.ext.security.management.keycloak.clientSecret=759514d0-dbb1-46ba-b7e7-ff76e63c6891
```



#### 注記

**WEB-INF/classes/security-management.properties** ファイルが存在しない場合は、そのファイルを作成します。

4. **/META-INF/jboss-deployment-structure.xml** ファイルにおいて、以下の依存関係および除外を編集します。

```
<dependencies>
```



```

    <module name="org.jboss.resteasy.resteasy-jackson-provider"
services="import"/>
</dependencies>
<exclusions>
    <module name="org.jboss.resteasy.resteasy-jackson2-provider"/>
</exclusions>

```

## 7.2. RH-SSO を使用した **DECISION SERVER** の認証

Red Hat Decision Manager の Decision Server は、サードパーティークライアントの REST API を提供します。Decision Server と RH-SSO を統合して、サードパーティークライアントのアイデンティティ管理を RH-SSO サーバーに委譲することができます。

Red Hat Decision Manager のレルムクライアントを作成して、Red Hat JBoss EAP の RH-SSO クライアントアダプターを設定したら、同じステップを繰り返して、Decision Server と RH-SSO を統合できます。

### 7.2.1. RH-SSO における **Decision Server** クライアントの作成

RH-SSO 管理コンソールを使用して、既存のレルムに新規クライアントを作成することができます。

#### 手順

1. RH-SSO 管理コンソールで作成したセキュリティーレルムを開きます。
2. **Clients** メニューアイテムをクリックし、**Create** をクリックします。  
**Add Client** ページが表示されます。
3. **Add Client** ページで、以下のようにレルム用の新規クライアント作成に必要な情報を指定します。
  - **Client ID**: kie-execution-server
  - **Root URL**: http://localhost:8080/kie-server
  - **Client protocol**: openid-connect
4. **Save** をクリックして変更を保存します。  
新規クライアントの **Access Type** は、デフォルトでは **public** に設定されています。この設定を **confidential** に変更して、もう一度 **Save** をクリックします。
5. **Credentials** タブに移動して秘密鍵をコピーします。秘密鍵は、次のセクションで kie-execution-server クライアントを設定するのに必要です。

### 7.2.2. クライアントアダプターおよび **Decision Server** のインストールおよび設定

Decision Server のリモートサービスエンドポイントを使用するには、最初に RH-SSO 管理コンソールで **kie-server** ロールを作成して割り当てる必要があります。



#### 注記

Decision Server を Red Hat Decision Manager とは別のアプリケーションサーバーにデプロイした場合には、第 2 サーバー にも RH-SSO をインストールして設定します。

## 手順

1. Red Hat JBoss EAP システムの **EAP\_HOME/standalone/configuration** に移動し、**standalone.xml** ファイルを編集して RH-SSO サブシステム設定を追加します。以下に例を示します。

```
<subsystem xmlns="urn:jboss:domain:keycloak:1.1">
  <secure-deployment name="kie-execution-server.war">
    <realm>demo</realm>
    <realm-public-
key>MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCrVrCuTtArbgaZzL1hvh0xtL5m
c7o0NqPVnYXkLvgcwiC3BjLGW1tGEGoJaXDuSaRllobm53JBhjx33UNv+5z/UMG4kytB
WxheNVKnL6Ggq1NabMaFfPLPCF8kAgKnsi79NMo+n6KnSY8YeUmec/p2vj02NjsSAVcW
EQMVhJ31LwIDAQAB</realm-public-key>
    <auth-server-url>http://localhost:8180/auth</auth-server-url>
    <ssl-required>external</ssl-required>
    <resource>kie-execution-server</resource>
    <enable-basic-auth>true</enable-basic-auth>
    <credential name="secret">03c2b267-7f64-4647-8566-
572be673f5fa</credential>
    <principal-attribute>preferred_username</principal-attribute>
  </secure-deployment>
</subsystem>

<system-properties>
  <property name="org.kie.server.sync.deploy" value="false"/>
</system-properties>
```

この例で、

- **secure-deployment name** は、アプリケーションの WAR ファイルの名前です。
  - **realm** は、使用するアプリケーション用に作成したレルムの名前です。
  - **realm-public-key** は、作成したレルムの公開鍵です。この鍵は、RH-SSO 管理コンソールで作成したレルムの **Realm settings** ページの **Keys** タブで確認することができます。この公開鍵の値を指定しない場合は、サーバーが自動的に取得します。
  - **auth-server-url** は、RH-SSO 認証サーバーの URL です。
  - **resource** は、作成したサーバークライアントの名前です。
  - **enable-basic-auth** は、クライアントがトークンベースと基本認証アプローチの両方を使用して要求を実行できるように、基本認証メカニズムを有効にする設定です。
  - **credential name** は、作成したサーバークライアントの秘密鍵です。この鍵は、RH-SSO 管理コンソールの **Clients** ページの **Credentials** タブで確認することができます。
  - **principal-attribute** は、ユーザーのログイン名です。この値を指定しないと、ユーザー名ではなくユーザー ID がアプリケーションに表示されます。
2. **standalone.xml** ファイルに加えた設定変更を保存します。
  3. 以下のコマンドを使用し、Red Hat JBoss EAP サーバーを再起動して Decision Server を実行します。

```
EXEC_SERVER_HOME/bin/standalone.sh -Dorg.kie.server.id=<ID> -
Dorg.kie.server.user=<USER> -Dorg.kie.server.pwd=<PWD> -
Dorg.kie.server.location=<LOCATION_URL> -Dorg.kie.server.controller=
<CONTROLLER_URL> -Dorg.kie.server.controller.user=<CONTROLLER_USER>
-Dorg.kie.server.controller.pwd=<CONTROLLER_PASSWORD>
```

以下に例を示します。

```
EXEC_SERVER_HOME/bin/standalone.sh -Dorg.kie.server.id=kieserver1 -
Dorg.kie.server.user=kieserver -Dorg.kie.server.pwd=password -
Dorg.kie.server.location=http://localhost:8080/kie-execution-
server/services/rest/server -
Dorg.kie.server.controller=http://localhost:8080/decision-
central/rest/controller -
Dorg.kie.server.controller.user=kiecontroller -
Dorg.kie.server.controller.pwd=password
```

4. Decision Server を実行したら、サーバーの状態を確認することができます。以下のコマンドの **kieserver** は **kie-server** ロールのユーザー名で、パスワードは **password** です。

```
curl http://kieserver:password@localhost:8080/kie-execution-
server/services/rest/server/
```

Red Hat Decision Manager と Decision Server 間の通信に、トークンベースの認証を使用することもできます。アプリケーションにおいて、(ユーザー名とパスワードではなく) 完全なトークンをアプリケーションサーバーのシステムプロパティーとして使用できます。ただし、トークンは自動的に更新されないため、アプリケーションの通信が行われている間トークンが失効しないようにする必要があります。トークンを取得する方法は「[トークンベースの認証](#)」を参照してください。

## 手順

1. トークンを使用して Decision Server を管理するように Red Hat Decision Manager を設定するには、**org.kie.server.token** プロパティーを設定します。
2. **org.kie.server.user** および **org.kie.server.pwd** プロパティーは設定しないでください。これで、Red Hat Decision Manager は **Authorization: Bearer \$TOKEN** 認証方法を使用します。

## 手順

1. トークンベースの認証を用いて REST API を使用する場合は、**org.kie.server.controller.token** プロパティーを設定します。
2. **org.kie.server.controller.user** および **org.kie.server.controller.pwd** プロパティーは設定しないでください。



## 注記

Decision Server はトークンを更新できないので、寿命が長いトークンを使用してください。トークンの有効期限は、2038 年 1 月 19 日以降にならないようにしてください。セキュリティのベストプラクティスをチェックし、お使いの環境に適したソリューションかどうかを確認してください。

## 7.3. RH-SSO を使用したサードパーティークライアントの認証

Red Hat Decision Manager または Decision Server が提供するさまざまなリモートサービスを使用するには、curl、wget、Web ブラウザー、またはカスタム REST クライアントなどのクライアントは、RH-SSO サーバー経由で認証して、要求を実行するための有効なトークンを取得する必要があります。リモートサービスを使用するには、認証済みのユーザーには以下のロールが割り当てられている必要があります。

- **rest-all**: Red Hat Decision Manager のリモートサービスを使用する場合
- **kie-server**: Decision Server のリモートサービスを使用する場合

RH-SSO 管理コンソールを使用してこれらのロールを作成し、リモートサービスを消費するユーザーに割り当てます。

クライアントは、以下のオプションのいずれかを使用して RH-SSO 経由で認証することができます。

- 基本認証 (クライアントがサポートしている場合)
- トークンベースの認証

### 7.3.1. 基本認証

RH-SSO クライアントアダプターの設定で Red Hat Decision Manager および Decision Server の両方に対する基本認証を有効にしている場合には、トークン付与/更新の呼び出しをせずにサービスを呼び出すことができます。以下に例を示します。

- Web ベースのリモートリポジトリエンドポイントの場合:

```
curl http://admin:password@localhost:8080/decision-
central/rest/repositories
```

- Decision Server の場合:

```
curl http://admin:password@localhost:8080/kie-execution-
server/services/rest/server/
```

### 7.3.2. トークンベースの認証

よりセキュアな認証オプションを希望される場合には、RH-SSO により付与されたトークンを使用して、Red Hat Decision Manager と Decision Server 両方のリモートサービスを使用できます。

#### 手順

1. RH-SSO 管理コンソールで **Clients** メニューアイテムをクリックし、**Create** をクリックして新規クライアントを作成します。  
**Add Client** ページが表示されます。
2. **Add Client** ページで、以下のようにレム用の新規クライアント作成に必要な情報を指定します。
  - **Client ID**: kie-remote
  - **Client protocol**: openid-connect

3. **Save** をクリックして変更を保存します。
4. **Realm Settings** でトークンの設定を変更します。
  - a. RH-SSO 管理コンソールで、**Realm Settings** メニューアイテムをクリックします。
  - b. **Tokens** タブをクリックします。
  - c. **Access Token Lifespan** の値を **15** 分に変更します。  
このように設定することで、トークンを取得してから失効するまでに十分な余裕をもってサービスを呼び出すことができます。
  - d. **Save** をクリックして変更を保存します。
5. リモートクライアントの公開クライアントを作成したら、以下のコマンドを使用して、RH-SSO サーバーのトークンエンドポイントに対して HTTP 要求を行いトークンを取得することができます。

```
RESULT=`curl --data "grant_type=password&client_id=kie-remote&username=admin&password=password"
http://localhost:8180/auth/realms/demo/protocol/openid-connect/token`
```

上記のコマンドは、RH-SSO ユーザーで実行してください。詳しい情報は「[新規ユーザーの追加](#)」を参照してください。

6. RH-SSO サーバーから取得したトークンを表示するには、以下のコマンドを使用します。

```
TOKEN=`echo $RESULT | sed 's/.*access_token": "/g' | sed 's/".*//g`
```

このトークンを使用してリモートの呼び出しを認証できるようになります。たとえば、Red Hat Decision Manager の内部リポジトリを確認するには、以下のようにトークンを使用します。

```
curl -H "Authorization: bearer $TOKEN" http://localhost:8080/decision-central/rest/repositories
```

## 7.4. LDAP および SSL の RED HAT DECISION MANAGER との統合

RH-SSO を用いて、LDAP および SSL を Red Hat Decision Manager と統合できます。LDAP および SSL を RH-SSO に対応させる設定については、『[Red Hat Single Sign-On Server Administration Guide](#)』を参照してください。

## 付録A バージョン情報

本ドキュメントの最終更新日: 2018 年 7 月 3 日