



Red Hat Process Automation Manager 7.5

Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.2

Red Hat Process Automation Manager 7.5 Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.2

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

Legal Notice

Copyright © 2020 Red Hat, Inc.

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

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

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

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

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

Linux[®] 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.

Abstract

This document describes how to install Red Hat Process Automation Manager on a Red Hat JBoss EAP 7.2 installation.

Table of Contents

PREFACE	4
CHAPTER 1. ABOUT RED HAT PROCESS AUTOMATION MANAGER	5
CHAPTER 2. RED HAT PROCESS AUTOMATION MANAGER ROLES AND USERS	6
CHAPTER 3. DOWNLOADING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION FILES	8
CHAPTER 4. USING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLER	9
4.1. USING THE INSTALLER IN INTERACTIVE MODE	9
4.2. USING THE INSTALLER IN CLI MODE	12
4.3. USING THE INSTALLER TO CONFIGURE PROCESS SERVER WITH THE PROCESS AUTOMATION MANAGER CONTROLLER	14
CHAPTER 5. INSTALLING RED HAT PROCESS AUTOMATION MANAGER FROM ZIP FILES	16
5.1. INSTALLING BUSINESS CENTRAL FROM THE ZIP FILE	16
5.2. INSTALLING PROCESS SERVER FROM THE ZIP FILE	17
5.3. CONFIGURING JDBC DATA SOURCES FOR PROCESS SERVER	18
5.4. CREATING USERS	20
5.5. CONFIGURING PROCESS SERVER WITH THE INTEGRATED PROCESS AUTOMATION MANAGER CONTROLLER	21
CHAPTER 6. SECURING PASSWORDS WITH A KEYSTORE	23
CHAPTER 7. CONFIGURING SSH TO USE RSA	25
CHAPTER 8. VERIFYING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION	26
CHAPTER 9. RUNNING RED HAT PROCESS AUTOMATION MANAGER	27
CHAPTER 10. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER	28
10.1. INSTALLING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER	28
10.1.1. Creating a headless Process Automation Manager controller user	29
10.1.2. Configuring Process Server and the headless Process Automation Manager controller	29
10.2. RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER	31
CHAPTER 11. RUNNING STANDALONE BUSINESS CENTRAL	33
11.1. BUSINESS CENTRAL SYSTEM PROPERTIES	34
CHAPTER 12. MAVEN SETTINGS AND REPOSITORIES FOR RED HAT PROCESS AUTOMATION MANAGER ..	40
12.1. ADDING MAVEN DEPENDENCIES FOR RED HAT PROCESS AUTOMATION MANAGER	40
12.2. CONFIGURING AN EXTERNAL MAVEN REPOSITORY FOR BUSINESS CENTRAL AND PROCESS SERVER	43
12.3. PREPARING A MAVEN MIRROR REPOSITORY FOR OFFLINE USE	45
CHAPTER 13. IMPORTING PROJECTS FROM GIT REPOSITORIES	47
CHAPTER 14. CUSTOMIZING THE BRANDING OF BUSINESS CENTRAL	48
14.1. CUSTOMIZING THE BUSINESS CENTRAL LOGIN PAGE	48
14.2. CUSTOMIZING BUSINESS CENTRAL APPLICATION HEADER	48
CHAPTER 15. INTEGRATING LDAP AND SSL	49
CHAPTER 16. ADDITIONAL RESOURCES	50

CHAPTER 17. NEXT STEPS	51
APPENDIX A. VERSIONING INFORMATION	52

PREFACE

This document describes how to install Red Hat Process Automation Manager on a Red Hat JBoss Enterprise Application Platform 7.2 instance.

Prerequisites

- You have reviewed the information in [Planning a Red Hat Process Automation Manager installation](#).
- You have installed the latest patch release of Red Hat JBoss Enterprise Application Platform 7.2. For information about installing Red Hat JBoss EAP, see the [Red Hat JBoss Enterprise Application Platform installation guide](#).

CHAPTER 1. ABOUT RED HAT PROCESS AUTOMATION MANAGER

Red Hat Process Automation Manager is the Red Hat middleware platform for creating business automation applications and microservices. It enables enterprise business and IT users to document, simulate, manage, automate, and monitor business processes and policies. It is designed to empower business and IT users to collaborate more effectively, so business applications can be changed easily and quickly.

Red Hat JBoss Enterprise Application Platform (Red Hat JBoss EAP) 7.2 is a certified implementation of the Java Enterprise Edition 8 (Java EE 8) full and web profile specifications. Red Hat JBoss EAP provides preconfigured options for features such as high availability, clustering, messaging, and distributed caching. It also enables users to write, deploy, and run applications using the various APIs and services that Red Hat JBoss EAP provides.

The instructions in this document explain how to install Red Hat Process Automation Manager in a Red Hat JBoss EAP 7.2 server instance.

For instruction on how to install Red Hat Process Automation Manager in other environments, see the following documents:

- [Installing and configuring Process Server on IBM WebSphere Application Server](#)
- [Installing and configuring Process Server on Oracle WebLogic Server](#)
- [Deploying a Red Hat Process Automation Manager immutable server environment on Red Hat OpenShift Container Platform](#)
- [Deploying a Red Hat Process Automation Manager authoring environment on Red Hat OpenShift Container Platform](#)
- [Deploying a Red Hat Process Automation Manager freeform managed server environment on Red Hat OpenShift Container Platform](#)
- [Deploying a Red Hat Process Automation Manager environment on Red Hat OpenShift Container Platform using Operators](#)

For information about supported components, see the following documents:

- [What is the mapping between Red Hat Process Automation Manager and the Maven library version?](#)
- [Red Hat Process Automation Manager 7 Supported Configurations](#)

CHAPTER 2. RED HAT PROCESS AUTOMATION MANAGER ROLES AND USERS

To access Business Central or Process Server, you must create users and assign them appropriate roles before the servers are started.

The Business Central and Process Server use Java Authentication and Authorization Service (JAAS) login module to authenticate the users. If both Business Central and Process Server are running on a single instance, then they share the same JAAS subject and security domain. Therefore, a user, who is authenticated for Business Central can also access Process Server.

However, if Business Central and Process Server are running on different instances, then the JAAS login module is triggered for both individually. Therefore, a user, who is authenticated for Business Central, needs to be authenticated separately to access the Process Server (for example, to view or manage process definitions in Business Central). In case, the user is not authenticated on the Process Server, then 401 error is logged in the log file, displaying **Invalid credentials to load data from remote server. Contact your system administrator.** message in Business Central.

This section describes available Red Hat Process Automation Manager user roles.



NOTE

The **admin**, **analyst**, **developer**, **manager**, **process-admin**, **user**, and **rest-all** roles are reserved for Business Central. The **kie-server** role is reserved for Process Server. For this reason, the available roles can differ depending on whether Business Central, Process Server, or both are installed.

- **admin:** Users with the **admin** role are the Business Central administrators. They can manage users and create, clone, and manage the repositories. They have full access to make required changes in the application. Users with the **admin** role have access to all areas within Red Hat Process Automation Manager.
- **analyst:** Users with the **analyst** role have access to all high-level features. They can model and execute their projects. However, these users cannot add contributors to spaces or delete spaces in the **Design → Projects** view. Access to the **Deploy → Execution Servers** view, which is intended for administrators, is not available to users with the **analyst** role. However, the **Deploy** button is available to these users when they access the Library perspective.
- **developer:** Users with the **developer** role have access to almost all features and can manage rules, models, process flows, forms, and dashboards. They can manage the asset repository, they can create, build, and deploy projects, and they can use Red Hat CodeReady Studio to view processes. Only certain administrative functions such as creating and cloning a new repository are hidden from users with the **developer** role.
- **manager:** Users with the **manager** role can view reports. These users are usually interested in statistics about the business processes and their performance, business indicators, and other business-related reporting. A user with this role has access only to process and task reports.
- **process-admin:** Users with the **process-admin** role are business process administrators. They have full access to business processes, business tasks, and execution errors. These users can also view business reports and have access to the Task Inbox list.
- **user:** Users with the **user** role can work on the Task Inbox list, which contains business tasks that are part of currently running processes. Users with this role can view process and task reports and manage processes.

- **rest-all**: Users with the **rest-all** role can access Business Central REST capabilities.
- **kie-server**: Users with the **kie-server** role can access Process Server (KIE Server) REST capabilities. This role is mandatory for users to have access to **Manage** and **Track** views in Business Central.

CHAPTER 3. DOWNLOADING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION FILES

Depending on your environment and installation requirements, download a Red Hat Process Automation Manager distribution.

Procedure

1. Navigate to the [Software Downloads](#) page in the Red Hat Customer Portal (login required), and select the product and version from the drop-down options:
 - **Product:** Process Automation Manager
 - **Version:** 7.5
2. Download one of the following product distributions, depending on your preferred installation method:



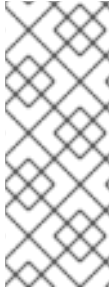
NOTE

You only need to download one of these distributions.

- If you want to use the installer to install Red Hat Process Automation Manager on Red Hat JBoss EAP 7.2 download **Red Hat Process Automation Manager 7.5.1 Installer**(**rhpm-installer-7.5.1.jar**). The installer graphical user interface guides you through the installation process.
- If you want to install Red Hat Process Automation Manager on Red Hat JBoss EAP 7.2 using the deployable ZIP files, download the following files:
 - **Red Hat Process Automation Manager 7.5.1 Process Server for All Supported EE8 Containers** (**rhpm-7.5.1-kie-server-ee8.zip**)
 - **Red Hat Process Automation Manager 7.5.1 Business Central Deployable for EAP 7** (**rhpm-7.5.1-business-central-eap7-deployable.zip**)
 - **Red Hat Process Automation Manager 7.5.1 Add Ons**(**rhpm-7.5.1-add-ons.zip**)
The ZIP files do not require a graphical user interface so you can install Red Hat Process Automation Manager using SSH.
- To run Business Central without needing to deploy it to an application server, download **Red Hat Process Automation Manager 7.5.1 Business Central Standalone** (**rhpm-7.5.1-business-central-standalone.jar**).

CHAPTER 4. USING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLER

This section describes how to install Process Server and the headless Process Automation Manager controller using the installer JAR file. The JAR file is an executable file that installs Red Hat Process Automation Manager in an existing Red Hat JBoss EAP 7.2 server installation. You can run the installer in interactive or command line interface (CLI) mode.



NOTE

The Red Hat Process Automation Manager JAR file installer does *not* support the Red Hat JBoss EAP distribution installed by yum or RPM Package Manager. If you want to install Red Hat Process Automation Manager in this type of Red Hat JBoss EAP installation, download the **Red Hat Process Automation Manager 7.5 Deployable for Red Hat JBoss EAP 7.2** file and follow the steps in [Chapter 5, Installing Red Hat Process Automation Manager from ZIP files](#).



NOTE

Because IBM JDK cannot use keystores generated on other JDKs, you cannot install Red Hat Process Automation Manager into an existing Red Hat JBoss EAP installation running on IBM JDK with a keystore generated on another JDK.

Next steps:

Follow the instructions in one of the following sections:

- [Section 4.1, “Using the installer in interactive mode”](#)
- [Section 4.2, “Using the installer in CLI mode”](#)
- [Section 4.3, “Using the installer to configure Process Server with the Process Automation Manager controller”](#)

4.1. USING THE INSTALLER IN INTERACTIVE MODE

The installer for Red Hat Process Automation Manager is an executable JAR file. You can use it to install Red Hat Process Automation Manager in an existing Red Hat JBoss EAP 7.2 server installation.



NOTE

For security reasons, you should run the installer as a non-root user.

Prerequisites

- A supported JDK is installed. For a list of supported JDKs, see [Red Hat Process Automation Manager 7 Supported Configurations](#).
- A backed-up Red Hat JBoss EAP 7.2 server installation is available.
- Sufficient user permissions to complete the installation are granted.
- The JAR binary is included in **\$PATH** environment variable. On Red Hat Enterprise Linux, it is included in the **java-\$JAVA_VERSION-openjdk-devel** package.



NOTE

Red Hat Process Automation Manager is designed to work with UTF-8 encoding. If a different encoding system is used by the underlying JVM, unexpected errors might occur. To ensure UTF-8 is used by the JVM, use the "**-Dfile.encoding=UTF-8**" system property.

Procedure

1. In a terminal window, navigate to the directory where you downloaded the installer JAR file and enter the following command:

```
java -jar rhpam-installer-7.5.1.jar
```



NOTE

When running the installer on Windows, you may be prompted to provide administrator credentials during the installation. To prevent this requirement, add the **izpack.mode=privileged** option to the installation command:

```
java -Dizpack.mode=privileged -jar  
rhpam-installer-7.5.1.jar
```

Furthermore, when running the installer on a 32-bit Java virtual machine, you might encounter memory limitations. To prevent this issue, run this command:

```
java -XX:MaxHeapSize=4g -jar  
rhpam-installer-7.5.1.jar
```

The graphical installer displays a splash screen and a license agreement page.

2. Click **I accept the terms of this license agreement** and click **Next**.
3. Specify the Red Hat JBoss EAP 7.2 server home where you want to install Red Hat Process Automation Manager and click **Next**.
4. Select the components that you want to install and click **Next**.



NOTE

You can install Business Central and Process Server on the same server. However, you should install Business Central and Process Server on different servers in production environments. To do this, run the installer twice.

5. Create a user and click **Next**. By default, if you install both Business Central and Process Server on the same server the new user is given the **admin**, **kie-server**, and **rest-all** roles. If you install only Process Server, the user is given the **kie-server** role. To select another role, deselect **admin**. For more information, see [Chapter 2, Red Hat Process Automation Manager roles and users](#).

**NOTE**

Make sure that the specified user name is not the same as an existing user, role, or group. For example, do not create a user with the user name **admin**.

The password must have at least eight characters and must contain at least one number and one non-alphanumeric character, but not & (ampersand).

Make a note of the user name and password. You will need them to access Business Central and Process Server.

6. On the **Installation Overview** page, click **Next** to start the installation. The Installation Overview page lists the components that you will install.
7. When the installation has completed, click **Next**.
8. On the **Configure Runtime Environment** page, choose whether to perform the default installation or perform an advanced configuration.
If you choose **Perform advanced configuration**, you can choose to configure database settings or customize certain Process Server options.
9. If you selected **Customize database settings**, on the **JDBC Drive Configuration** page specify a data source JDBC driver vendor, select one or more Driver JAR files, and click **Next**.
A data source is an object that enables a Java Database Connectivity (JDBC) client, such as an application server, to establish a connection with a database. Applications look up the data source on the Java Naming and Directory Interface (JNDI) tree or in the local application context and request a database connection to retrieve data. You must configure data sources for Process Server to ensure proper data exchange between the servers and the designated database.
10. If you selected **Customize Process Server settings**, change any of the following if required:
 - Change the name of the Process Server property.
 - Change the URL of the headless Process Automation Manager controller.
 - Deselect any Process Server server options.
11. Click **Next** to configure the runtime environment.
12. When **Processing finished** appears at the top of the screen, click **Next** to complete the installation.
13. Optionally, click **Generate Installation Script and Properties File** to save the installation data in XML files, and then click **Done**.
The installer generates two files. The **auto.xml** file automates future installations and the **auto.xml.variables** file stores user passwords and other sensitive variables. Use the **auto.xml** file on multiple systems to easily repeat a Red Hat Process Automation Manager installation on the same type of server with the same configuration as the original installation. If necessary, update the **installpath** parameter in the **auto.xml** file. Enter the following command to perform an installation with the XML file:

```
java -jar rhpam-installer-7.5.1.jar <path-to-auto.xml-file>
```

You have successfully installed Red Hat Process Automation Manager using the installer. If you installed only Business Central, repeat these steps to install Process Server on a separate server.

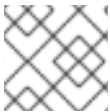
**NOTE**

If you use Microsoft SQL Server, make sure you have configured proper transaction isolation for your database. If you do not, you may experience deadlocks. The recommended configuration is to turn on `ALLOW_SNAPSHOT_ISOLATION` and `READ_COMMITTED_SNAPSHOT` by entering the following statements:

```
ALTER DATABASE <DBNAME> SET ALLOW_SNAPSHOT_ISOLATION ON
ALTER DATABASE <DBNAME> SET READ_COMMITTED_SNAPSHOT ON
```

4.2. USING THE INSTALLER IN CLI MODE

You can run the Red Hat Process Automation Manager installer through the command-line interface (CLI).

**NOTE**

For security reasons, you should run the installer as a non-root user.

Prerequisites

- A supported JDK is installed. For a list of supported JDKs, see [Red Hat Process Automation Manager 7 Supported Configurations](#).
- A backed-up Red Hat JBoss EAP 7.2 server installation is available.
- Sufficient user permissions to complete the installation are granted.
- The JAR binary is included in the `$PATH` environment variable. On Red Hat Enterprise Linux, it is included in the `java-$JAVA_VERSION-openjdk-devel` package.

**NOTE**

Red Hat Process Automation Manager is designed to work with UTF-8 encoding. If a different encoding system is used by the underlying JVM, unexpected errors might occur. To ensure UTF-8 is used by the JVM, use the "`-Dfile.encoding=UTF-8`" system property.

Procedure

1. In a terminal window, navigate to the directory where you downloaded the installer file and enter the following command:

```
java -jar rhpam-installer-7.5.1.jar -console
```

The command-line interactive process will start and display the End-User License Agreement.

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

2. Read the license agreement, enter **1**, and press Enter to continue:

```
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. Enter the parent directory of an existing Red Hat JBoss EAP 7.2 installation. The installer will verify the location of the installation at the location provided. Enter **1** to confirm and continue.

**NOTE**

You can install Business Central and Process Server on the same server. However, you should install Business Central and Process Server on different servers in production environments.

4. Follow the instructions in the installer to complete the installation.

**NOTE**

When you create the user name and password, make sure that the specified user name does not conflict with any known title of a role or a group. For example, if there is a role called **admin**, you should not create a user with the user name **admin**.

The password must have at least eight characters and must contain at least one number and one non-alphanumeric character (*not* including the character **&**).

Make a note of the user name and password. You will need them to access Business Central and Process Server.

5. When the installation has completed, you will see this message:

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

6. Enter **y** to create XML files that contain the installation data, or **n** to complete the installation. If you enter **y**, you are prompted to specify a path for the XML files.
7. Enter a path or press the Enter key to accept the suggested path. The installer generates two files. The **auto.xml** file automates future installations and the **auto.xml.variables** file stores user passwords and other sensitive variables. Use the **auto.xml** file on multiple systems to easily repeat a Red Hat Process Automation Manager installation on the same type of server with the same configuration as the original installation. If necessary, update the **installpath** parameter in the **auto.xml** file. Enter the following command to perform an installation with the XML file:

```
java -jar rhpam-installer-7.5.1.jar <path-to-auto.xml-file>
```

8. If you installed only Business Central, repeat these steps to install Process Server on a separate server.

**NOTE**

If you use Microsoft SQL Server, make sure you have configured proper transaction isolation for your database. If you do not, you may experience deadlocks. The recommended configuration is to turn on `ALLOW_SNAPSHOT_ISOLATION` and `READ_COMMITTED_SNAPSHOT` by entering the following statements:

```
ALTER DATABASE <DBNAME> SET ALLOW_SNAPSHOT_ISOLATION ON
ALTER DATABASE <DBNAME> SET READ_COMMITTED_SNAPSHOT ON
```

4.3. USING THE INSTALLER TO CONFIGURE PROCESS SERVER WITH THE PROCESS AUTOMATION MANAGER CONTROLLER

Process Server can be managed by the Process Automation Manager controller or it can be unmanaged. If Process Server is unmanaged, you must manually create and maintain KIE containers (deployment units). If Process Server is managed, the Process Automation Manager controller manages the Process Server configuration and you interact with the Process Automation Manager controller to create and maintain KIE containers.

The Process Automation Manager controller is integrated with Business Central. If you install Business Central, you can use the **Execution Server** page in Business Central to interact with the Process Automation Manager controller.

You can use the installer in interactive or CLI mode to install Business Central and Process Server, and then configure Process Server with the Process Automation Manager controller.

**NOTE**

If you do not install Business Central, see [Chapter 10, *Installing and running the headless Process Automation Manager controller*](#) for information about using the headless Process Automation Manager controller.

Prerequisites

- Two computers with backed-up Red Hat JBoss EAP 7.2 server installations are available.
- Sufficient user permissions to complete the installation are granted.

Procedure

1. On the first computer, run the installer in interactive mode or CLI mode. See one of the following sections for more information:
 - [Section 4.1, "Using the installer in interactive mode"](#)
 - [Section 4.2, "Using the installer in CLI mode"](#)
2. On the **Component Selection** page, clear the **Process Server** box.
3. Complete the Business Central installation.
4. On the second computer, run the installer in interactive mode or CLI mode.
5. On the **Component Selection** page, clear the **Business Central** box.

6. On the **Configure Runtime Environment** page, select **Perform Advanced Configuration**.
7. Select **Customize Process Server properties** and click **Next**.
8. Enter the controller URL for Business Central and configure additional properties for Process Server. The controller URL has the following form where **<HOST:PORT>** is the address of Business Central on the second computer:


```
<HOST:PORT>/business-central/rest/controller
```
9. Complete the installation.
10. To verify that the Process Automation Manager controller is now integrated with Business Central, go to the **Execution Servers** page in Business Central and confirm that the Process Server that you configured appears under **REMOTE SERVERS**.

CHAPTER 5. INSTALLING RED HAT PROCESS AUTOMATION MANAGER FROM ZIP FILES

The Red Hat Process Automation Manager ZIP files (one for Business Central and one for Process Server) do not require a graphical user interface so you can install Red Hat Process Automation Manager using SSH.



NOTE

You should install Business Central and the Process Server on different servers in production environments.

For information about installing the headless Process Automation Manager controller, see [Chapter 10, *Installing and running the headless Process Automation Manager controller*](#).

5.1. INSTALLING BUSINESS CENTRAL FROM THE ZIP FILE

Business Central is a web console that enables you to perform the following tasks over individual components in a unified web-based environment:

- Create, manage, and edit your rules, processes, and related assets.
- Manage connected Process Server instances and their KIE containers (deployment units).
- Execute runtime operations against processes and tasks in Process Server instances connected to Business Central.

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.2 is available. The base directory of the Red Hat JBoss EAP installation is referred to as **EAP_HOME**.
- Sufficient user permissions to complete the installation are granted.
- The following file is downloaded as described in [Chapter 3, *Downloading the Red Hat Process Automation Manager installation files*](#):
rhpmam-7.5.1-business-central-eap7-deployable.zip

Procedure

1. Extract the **rhpmam-7.5.1-business-central-eap7-deployable.zip** file to a temporary directory. In the following examples this directory is called **TEMP_DIR**.
2. Copy the contents of the **TEMP_DIR/rhpmam-7.5.1-business-central-eap7-deployable/jboss-eap-7.2** directory to **EAP_HOME**. When prompted, merge or replace files.

**WARNING**

Ensure the names of the Red Hat Process Automation Manager deployments you copy do not conflict with your existing deployments in the Red Hat JBoss EAP instance.

5.2. INSTALLING PROCESS SERVER FROM THE ZIP FILE

Process Server provides the runtime environment for business assets and accesses the data stored in the assets repository (knowledge store).

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.2 is available. The base directory of the Red Hat JBoss EAP installation is referred to as **EAP_HOME**.
- Sufficient user permissions to complete the installation are granted.
- The following file is downloaded as described in [Chapter 3, Downloading the Red Hat Process Automation Manager installation files](#):
rhpm-7.5.1-kie-server-ee8.zip

Procedure

1. Extract the **rhpm-7.5.1-kie-server-ee8.zip** archive to a temporary directory. In the following examples this directory is called **TEMP_DIR**.
2. Copy the **TEMP_DIR/rhpm-7.5.1-kie-server-ee8/kie-server.war** directory to **EAP_HOME/standalone/deployments/**.

**WARNING**

Ensure the names of the Red Hat Process Automation Manager deployments you copy do not conflict with your existing deployments in the Red Hat JBoss EAP instance.

3. Copy the contents of the **TEMP_DIR/rhpm-7.5.1-kie-server-ee8/rhpm-7.5.1-kie-server-ee8/SecurityPolicy/** to **EAP_HOME/bin**. When asked to overwrite files, click **Replace**.
4. In the **EAP_HOME/standalone/deployments/** directory, create an empty file named **kie-server.war.dodeploy**. This file ensures that Process Server is automatically deployed when the server starts.



NOTE

If you use Microsoft SQL Server, make sure you have configured proper transaction isolation for your database. If you do not, you may experience deadlocks. The recommended configuration is to turn on `ALLOW_SNAPSHOT_ISOLATION` and `READ_COMMITTED_SNAPSHOT` by entering the following statements:

```
ALTER DATABASE <DBNAME> SET ALLOW_SNAPSHOT_ISOLATION ON
ALTER DATABASE <DBNAME> SET READ_COMMITTED_SNAPSHOT ON
```

5.3. CONFIGURING JDBC DATA SOURCES FOR PROCESS SERVER

A data source is an object that enables a Java Database Connectivity (JDBC) client, such as an application server, to establish a connection with a database. Applications look up the data source on the Java Naming and Directory Interface (JNDI) tree or in the local application context and request a database connection to retrieve data. You must configure data sources for Process Server to ensure proper data exchange between the servers and the designated database.



NOTE

For production environments, specify an actual data source. Do not use the example data source in production environments.

Prerequisites

- The JDBC providers that you want to use to create database connections are configured on all servers on which you want to deploy Process Server, as described in the "Creating Datasources" and "JDBC Drivers" sections of the [Red Hat JBoss Enterprise Application Server Configuration Guide](#).
- The Red Hat Process Automation Manager 7.5.1 Add Ons (rhpmam-7.5.1-add-ons.zip) file is downloaded from the [Software Downloads](#) page in the Red Hat Customer Portal.

Procedure

1. Complete the following steps to prepare your database:
 - a. Extract **rhpmam-7.5-add-ons.zip** in a temporary directory, for example **TEMP_DIR**.
 - b. Extract **TEMP_DIR/rhpmam-7.5-migration-tool.zip**.
 - c. Change your current directory to the **TEMP_DIR/rhpmam-7.5-migration-tool/ddl-scripts** directory. This directory contains DDL scripts for several database types.
 - d. Import the DDL script for your database type into the database that you want to use, for example:

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

2. Open **EAP_HOME/standalone/configuration/standalone-full.xml** in a text editor and locate the **<system-properties>** tag.

3. Add the following properties to the `<system-properties>` tag where `<DATASOURCE>` is the JNDI name of your data source and `<HIBERNATE_DIALECT>` is the hibernate dialect for your database.



NOTE

The default value of the `org.kie.server.persistence.ds` property is `java:jboss/datasources/ExampleDS`. The default value of the `org.kie.server.persistence.dialect` property is `org.hibernate.dialect.H2Dialect`.

```
<property name="org.kie.server.persistence.ds" value="<DATASOURCE>"/>
<property name="org.kie.server.persistence.dialect" value="<HIBERNATE_DIALECT>"/>
```

The following example shows how to configure a datasource for the PostgreSQL hibernate dialect:

```
<system-properties>
  <property name="org.kie.server.repo" value="${jboss.server.data.dir}"/>
  <property name="org.kie.example" value="true"/>
  <property name="org.jbpm.designer.perspective" value="full"/>
  <property name="designerdataobjects" value="false"/>
  <property name="org.kie.server.user" value="rhpamUser"/>
  <property name="org.kie.server.pwd" value="rhpam123!"/>
  <property name="org.kie.server.location" value="http://localhost:8080/kie-
server/services/rest/server"/>
  <property name="org.kie.server.controller" value="http://localhost:8080/business-
central/rest/controller"/>
  <property name="org.kie.server.controller.user" value="kieserver"/>
  <property name="org.kie.server.controller.pwd" value="kieserver1!"/>
  <property name="org.kie.server.id" value="local-server-123"/>

  <!-- Data source properties. -->
  <property name="org.kie.server.persistence.ds"
value="java:jboss/datasources/KieServerDS"/>
  <property name="org.kie.server.persistence.dialect"
value="org.hibernate.dialect.PostgreSQLDialect"/>
</system-properties>
```

The following dialects are supported:

- DB2: `org.hibernate.dialect.DB2Dialect`
- MSSQL: `org.hibernate.dialect.SQLServer2012Dialect`
- MySQL: `org.hibernate.dialect.MySQL5InnoDBDialect`
- MariaDB: `org.hibernate.dialect.MySQL5InnoDBDialect`
- Oracle: `org.hibernate.dialect.Oracle10gDialect`
- PostgreSQL: `org.hibernate.dialect.PostgreSQL82Dialect`
- PostgreSQL plus: `org.hibernate.dialect.PostgresPlusDialect`
- Sybase: `org.hibernate.dialect.SybaseASE157Dialect`

5.4. CREATING USERS

Before you can log in to Business Central or Process Server, you must create users. This section shows you how to create a Business Central user with the **admin**, **rest-all**, and **kie-server** roles and a Process Server user that has the **kie-server** role. For information about roles, see [Chapter 2, Red Hat Process Automation Manager roles and users](#).

Prerequisites

- Red Hat Process Automation Manager is installed in the base directory of the Red Hat JBoss EAP installation (**EAP_HOME**).

Procedure

- In a terminal application, navigate to the **EAP_HOME/bin** directory.
- Create a user with the **admin**, **rest-all**, and **kie-server** roles that you will use to log in to Business Central.



NOTE

Users with the **admin** role are Business Central administrators. Users with **rest-all** role can access Business Central REST capabilities. Users with the **kie-server** role can access Process Server (KIE Server) REST capabilities. The **kie-server** role is mandatory for users to have access to **Manage** and **Track** views in Business Central.

In the following command, replace **<username>** and **<password>** with the user name and password of your choice.

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



NOTE

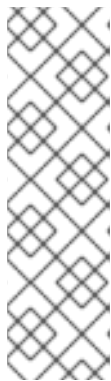
Make sure that the specified user name is not the same as an existing user, role, or group. For example, do not create a user with the user name **admin**.

The password must have at least eight characters and must contain at least one number and one non-alphanumeric character, but not & (ampersand).

- Create a user with the **kie-server** role that you will use to log in to Process Server.

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

- Make a note of your user names and passwords.

**NOTE**

If you installed Business Central and Process Server in the same server instance, you can create a single user that has both of these roles:

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

You should install Business Central and Process Server on different servers in production environments.

5.5. CONFIGURING PROCESS SERVER WITH THE INTEGRATED PROCESS AUTOMATION MANAGER CONTROLLER

**NOTE**

Only make the changes described in this section if Process Server will be managed by Business Central and you installed Red Hat Process Automation Manager from the ZIP files. If you did not install Business Central, you can use the headless Process Automation Manager controller to manage Process Server, as described in [Chapter 10, *Installing and running the headless Process Automation Manager controller*](#).

Process Server can be managed or it can be unmanaged. If Process Server is unmanaged, you must manually create and maintain KIE containers (deployment units). If Process Server is managed, the Process Automation Manager controller manages the Process Server configuration and you interact with the Process Automation Manager controller to create and maintain KIE containers.

The Process Automation Manager controller is integrated with Business Central. If you install Business Central, you can use the **Execution Server** page in Business Central to interact with the Process Automation Manager controller.

If you installed Red Hat Process Automation Manager from the ZIP files, you must edit the **standalone-full.xml** file in both the Process Server and Business Central installations to configure Process Server with the integrated Process Automation Manager controller.

Prerequisites

- Business Central and Process Server are installed in the base directory of the Red Hat JBoss EAP installation (**EAP_HOME**) as described in the following sections:
 - [Section 5.1, “Installing Business Central from the ZIP file”](#)
 - [Section 5.2, “Installing Process Server from the ZIP file”](#)

**NOTE**

You should install Business Central and Process Server on different servers in production environments. However, if you install Process Server and Business Central on the same server, for example in a development environment, make the changes described in this section in the shared **standalone-full.xml** file.

- On Business Central server nodes, a user with the **rest-all** role exists.

For more information, see [Section 5.4, "Creating users"](#).

Procedure

1. In the Business Central `EAP_HOME/standalone/configuration/standalone-full.xml` file, uncomment the following properties in the `<system-properties>` section and replace `<USERNAME>` and `<USER_PWD>` with the credentials of a user with the `kie-server` role:

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

2. In the Process Server `EAP_HOME/standalone/configuration/standalone-full.xml` file, uncomment the following properties in the `<system-properties>` section.

```
<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>"/>
```

3. Replace the following values:
 - Replace `<CONTROLLER_USER>` and `<CONTROLLER_PWD>` with the credentials of a user with the `rest-all` role.
 - Replace `<KIE_SERVER_ID>` with the ID or name of the Process Server installation, for example, `rh pam-7.5.1-process_server-1`.
 - Replace `<HOST>` with the ID or name of the Process Server host, for example, `localhost` or `192.7.8.9`.
 - Replace `<PORT>` with the port of the Process Server host, for example, `8080`.



NOTE

The `org.kie.server.location` property specifies the location of Process Server.

- Replace `<CONTROLLER_URL>` with the URL of Business Central. Process Server connects to this URL during startup.
 - If you installed Business Central using the installer or Red Hat JBoss EAP zip installations, `<CONTROLLER_URL>` has this format:
http://<HOST>:<PORT>/business-central/rest/controller
 - If you are running Business Central using the `standalone.jar` file, `<CONTROLLER_URL>` has this format:
http://<HOST>:<PORT>/rest/controller

CHAPTER 6. SECURING PASSWORDS WITH A KEYSTORE

You can use a keystore to encrypt passwords that are used for communication between Business Central and Process Server. You should encrypt both controller and Process Server passwords. If Business Central and Process Server are deployed to different application servers, then both application servers should use the keystore.

Use Java Cryptography Extension KeyStore (JCEKS) for your keystore because it supports symmetric keys. Use KeyTool, which is part of the JDK installation, to create a new JCEKS.



NOTE

If Process Server is not configured with JCEKS, Process Server passwords are stored in system properties in plain text form.

Prerequisites

- Process Server is installed in Red Hat JBoss EAP.
- Java 8 or higher is installed.

Procedure

1. In the Red Hat JBoss EAP home directory, enter the following command to create a Process Server user with the **kie-server** role and specify a password. In the following example, replace **<USER_NAME>** and **<PASSWORD>** with the user name and password of your choice.

```
$<EAP_HOME>/bin/add-user.sh -a -e -u <USER_NAME> -p <PASSWORD> -g kie-server
```

2. To use KeyTool to create a JCEKS, enter the following command in the Java 8 home directory:

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

In this example, replace the following variables:

- **<KEYSTORE_PATH>**: The path where the keystore will be stored
 - **<KEYSTORE_PASSWORD>**: The keystore password
 - **<ALIAS_KEY_PASSWORD>**: The password used to access values stored with the alias
 - **<PASSWORD_ALIAS>**: The alias of the entry to the process
3. When prompted, enter the password for the Process Server user that you created.
 4. Set the following system properties in the **EAP_HOME/standalone/configuration/standalone-full.xml** file and replace the placeholders as listed in the following table:

```
<system-properties>
  <property name="kie.keystore.keyStoreURL" value="<KEYSTORE_URL>"/>
  <property name="kie.keystore.keyStorePwd" value="<KEYSTORE_PWD>"/>
  <property name="kie.keystore.key.server.alias" value="<KEY_SERVER_ALIAS>"/>
  <property name="kie.keystore.key.server.pwd" value="<KEY_SERVER_PWD>"/>
</system-properties>
```

```

<property name="kie.keystore.key.ctrl.alias" value="<KEY_CONTROL_ALIAS>"/>
<property name="kie.keystore.key.ctrl.pwd" value="<KEY_CONTROL_PWD>"/>
</system-properties>

```

Table 6.1. System properties used to load a Process Server JCEKS

System property	Placeholder	Description
kie.keystore.keyStoreURL	<KEYSTORE_URL>	URL for the JCEKS that you want to use, for example file:///home/kie/keystores/keystore.jceks
kie.keystore.keyStorePwd	<KEYSTORE_PWD>	Password for the JCEKS
kie.keystore.key.server.alias	<KEY_SERVER_ALIASES>	Alias of the key for REST services where the password is stored
kie.keystore.key.server.pwd	<KEY_SERVER_PWD>	Password of the alias for REST services with the stored password
kie.keystore.key.ctrl.alias	<KEY_CONTROL_ALIASES>	Alias of the key for default REST Process Automation Controller where the password is stored
kie.keystore.key.ctrl.key.ctrl.pwd	<KEY_CONTROL_PWD>	Password of the alias for default REST Process Automation Controller with the stored password

5. Start Process Server to verify the configuration.

CHAPTER 7. CONFIGURING SSH TO USE RSA

SSH is used to clone Git repositories. By default, the DSA encryption algorithm is provided by Business Central. However, some SSH clients, for example SSH clients in the Fedora 23 environment, use the RSA algorithm instead of the DSA algorithm. Business Central contains a system property that you can use to switch from DSA to RSA if required.



NOTE

SSH clients on supported configurations, for example Red Hat Enterprise Linux 7, are not affected by this issue. For a list of supported configurations, see [Red Hat Process Automation Manager 7 Supported Configurations](#).

Procedure

Complete one of the following tasks to enable this system property:

- Modify the `~/.ssh/config` file on client side as follows to force the SSH client to accept the deprecated DSA algorithm:

```
Host <SERVER_IP>  
  HostKeyAlgorithms +ssh-dss
```

- Include the `-Dorg.uberfire.nio.git.ssh.algorithm=RSA` parameter when you start Business Central, for example:

```
$ ./standalone.sh -c standalone-full.xml  
-Dorg.uberfire.nio.git.ssh.algorithm=RSA
```

CHAPTER 8. VERIFYING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION

After you have installed Red Hat Process Automation Manager, create an asset to verify that the installation is working.

Procedure

1. In a terminal window, navigate to the **EAP_HOME/bin** directory and enter the following command to start Red Hat Process Automation Manager:

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



NOTE

If you deployed Business Central without Process Server, you can start Business Central with the **standalone.sh** script without specifying the **standalone-full.xml** file. In this case, ensure that you make any configuration changes to the **standalone.xml** file before starting Business Central.

2. In a web browser, enter **localhost:8080/business-central**.
 - If Red Hat Process Automation Manager has been configured to run from a domain name, replace **localhost** with the domain name, for example:
http://www.example.com:8080/business-central
 - If Red Hat Process Automation Manager has been configured to run in a cluster, replace **localhost** with the IP address of a particular node, for example:
http://<node_IP_address>:8080/business-central
3. Enter the **admin** user credentials that you created during installation. The Business Central home page appears.
4. Select **Menu** → **Deploy** → **Execution Servers**.
5. Confirm that **default-kieserver** is listed under **Server Configurations**.
6. Select **Menu** → **Design** → **Projects** → **MySpace**.
7. Click **Try Samples** → **Mortgages** → **OK**. The **Assets** window appears.
8. Click **Add Asset** → **Data Object**.
9. Enter **MyDataObject** in the **Data Object** field and click **OK**.
10. Click **Spaces** → **MySpace** → **Mortgages** and confirm that **MyDataObject** is in the list of assets.
11. Delete the **Mortgages** project.

CHAPTER 9. RUNNING RED HAT PROCESS AUTOMATION MANAGER

Use this procedure to run the Red Hat Process Automation Manager on Red Hat JBoss EAP in standalone mode.

Prerequisites

- Red Hat Process Automation Manager is installed and configured.



NOTE

If you changed the default host (**localhost**) or the default port (**9990**), then before you run Red Hat Process Automation Manager, you must edit the **business-central.war/WEB-INF/classes/datasource-management.properties** and **business-central.war/WEB-INF/classes/security-management.properties** files as described in [Solution 3519551](#).

Procedure

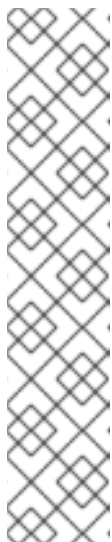
1. In a terminal application, navigate to **EAP_HOME/bin**.
2. Run the standalone configuration:

- On Linux or UNIX-based systems:

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

- On Windows:

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



NOTE

If you deployed Business Central without Process Server, you can start Business Central with the **standalone.sh** script without specifying the **standalone-full.xml** file. In this case, ensure that you make any configuration changes to the **standalone.xml** file before starting Business Central.

On Linux or UNIX-based systems:

```
./standalone.sh
```

On Windows:

```
standalone.bat
```

3. In a web browser, open the URL **localhost:8080/business-central**.
4. Log in using the credentials of the user that you created for Business Central in [Section 5.4, "Creating users"](#).

CHAPTER 10. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

You can configure Process Server to run in managed or unmanaged mode. If Process Server is unmanaged, you must manually create and maintain KIE containers (deployment units). If Process Server is managed, the Process Automation Manager controller manages the Process Server configuration and you interact with the Process Automation Manager controller to create and maintain KIE containers.

Business Central has an embedded Process Automation Manager controller. If you install Business Central, use the **Execution Server** page to create and maintain KIE containers. If you want to automate Process Server management without Business Central, you can use the headless Process Automation Manager controller.

10.1. INSTALLING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

You can install the headless Process Automation Manager controller and use the REST API or the Process Server Java Client API to interact with it.

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.2 is available. The base directory of the Red Hat JBoss EAP installation is referred to as **EAP_HOME**.
- Sufficient user permissions to complete the installation are granted.

Procedure

1. Navigate to the [Software Downloads](#) page in the Red Hat Customer Portal (login required), and select the product and version from the drop-down options:
 - **Product:** Process Automation Manager
 - **Version:** 7.5
2. Download **Red Hat Process Automation Manager 7.5.1 Add Ons**(the **rhpmam-7.5.1-add-ons.zip** file).
3. Unzip the **rhpmam-7.5.1-add-ons.zip** file. The **rhpmam-7.5-controller-ee7.zip** file is in the unzipped directory.
4. Extract the **rhpmam-7.5-controller-ee7** archive to a temporary directory. In the following examples this directory is called **TEMP_DIR**.
5. Copy the **TEMP_DIR/rhpmam-7.5-controller-ee7/controller.war** directory to **EAP_HOME/standalone/deployments/**.

**WARNING**

Ensure that the names of the headless Process Automation Manager controller deployments you copy do not conflict with your existing deployments in the Red Hat JBoss EAP instance.

6. Copy the contents of the ***TEMP_DIR*/rhpam-7.5-controller-ee7/SecurityPolicy/** directory to ***EAP_HOME*/bin**. When asked to overwrite files, select **Yes**.
7. In the ***EAP_HOME*/standalone/deployments/** directory, create an empty file named **controller.war.dodeploy**. This file ensures that the headless Process Automation Manager controller is automatically deployed when the server starts.

10.1.1. Creating a headless Process Automation Manager controller user

Before you can use the headless Process Automation Manager controller, you must create a user that has the **kie-server** role.

Prerequisites

- The headless Process Automation Manager controller is installed in the base directory of the Red Hat JBoss EAP installation (***EAP_HOME***).

Procedure

1. In a terminal application, navigate to the ***EAP_HOME*/bin** directory.
2. Enter the following command and replace **<USER_NAME>** and **<PASSWORD>** with the user name and password of your choice.

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

**NOTE**

Make sure that the specified user name is not the same as an existing user, role, or group. For example, do not create a user with the user name **admin**.

The password must have at least eight characters and must contain at least one number and one non-alphanumeric character, but not & (ampersand).

3. Make a note of your user name and password.

10.1.2. Configuring Process Server and the headless Process Automation Manager controller

If Process Server will be managed by the headless Process Automation Manager controller, you must edit the **standalone-full.xml** file in Process Server installation and the **standalone.xml** file in the headless Process Automation Manager controller installation, as described in this section.

Prerequisites

- Process Server is installed in the base directory of the Red Hat JBoss EAP installation (**EAP_HOME**) as described in [Chapter 5, Installing Red Hat Process Automation Manager from ZIP files](#) section.
- The headless Process Automation Manager controller is installed in an **EAP_HOME**.



NOTE

You should install Process Server and the headless Process Automation Manager controller on different servers in production environments. However, if you install Process Server and the headless Process Automation Manager controller on the same server, for example in a development environment, make these changes in the shared **standalone-full.xml** file.

- On Process Server nodes, a user with the **kie-server** role exists.
- On the server nodes, a user with the **kie-server** role exists.
For more information, see [Section 5.4, "Creating users"](#).

Procedure

1. In the **EAP_HOME/standalone/configuration/standalone-full.xml** file, add the following properties to the **<system-properties>** section and replace **<USERNAME>** and **<USER_PWD>** with the credentials of a user with the **kie-server** role:

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

2. In the Process Server **EAP_HOME/standalone/configuration/standalone-full.xml** file, add the following properties to the **<system-properties>** section:

```
<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>"/>
```

3. In this file, replace the following values:
 - Replace **<CONTROLLER_USER>** and **<CONTROLLER_PWD>** with the credentials of a user with the **kie-server** role.
 - Replace **<KIE_SERVER_ID>** with the ID or name of the Process Server installation, for example, **rhcam-7.5.1-process_server-1**.
 - Replace **<HOST>** with the ID or name of the Process Server host, for example, **localhost** or **192.7.8.9**.
 - Replace **<PORT>** with the port of the Process Server host, for example, **8080**.

**NOTE**

The **org.kie.server.location** property specifies the location of Process Server.

- Replace **<CONTROLLER_URL>** with the URL of the headless Process Automation Manager controller.
4. Process Server connects to this URL during startup.

10.2. RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

After you have installed the headless Process Automation Manager controller on Red Hat JBoss EAP, use this procedure to run the headless Process Automation Manager controller.

Prerequisites

- The headless Process Automation Manager controller is installed and configured in the base directory of the Red Hat JBoss EAP installation (**EAP_HOME**).

Procedure

1. In a terminal application, navigate to **EAP_HOME/bin**.
2. If you installed the headless Process Automation Manager controller on the same Red Hat JBoss EAP instance as the Red Hat JBoss EAP instance where you installed the Process Server, enter one of the following commands:
 - On Linux or UNIX-based systems:


```
┆ $ ./standalone.sh -c standalone-full.xml
```
 - On Windows:


```
┆ standalone.bat -c standalone-full.xml
```
3. If you installed the headless Process Automation Manager controller on a separate Red Hat JBoss EAP instance from the Red Hat JBoss EAP instance where you installed the Process Server, you can start the headless Process Automation Manager controller with the **standalone.sh** script:

**NOTE**

In this case, ensure that you made all required configuration changes to the **standalone.xml** file.

- On Linux or UNIX-based systems:

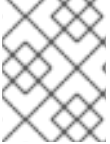
```
┆ $ ./standalone.sh
```

- On Windows:

```
standalone.bat
```

4. To verify that the headless Process Automation Manager controller is working on Red Hat JBoss EAP, enter the following command where **<CONTROLLER>** and **<CONTROLLER_PWD>** is the user name and password combination that you created in [Section 10.1.1, "Creating a headless Process Automation Manager controller user"](#) . The output of this command provides information about the Process Server instance.

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



NOTE

Alternatively, you can use the Process Server Java API Client to access the headless Process Automation Manager controller.

CHAPTER 11. RUNNING STANDALONE BUSINESS CENTRAL

You can use the Business Central standalone JAR file to run Business Central without needing to deploy it to an application server such as Red Hat JBoss EAP.



NOTE

Red Hat supports this installation type only when it is installed on Red Hat Enterprise Linux.

Prerequisites

- The **Red Hat Process Automation Manager 7.5.1 Business Central Standalone**(**rhpm-7.5.1-business-central-standalone.jar**) file has been downloaded, as described in [Chapter 3, Downloading the Red Hat Process Automation Manager installation files](#) .

Procedure

1. Create a directory and move the **rhpm-7.5.1-business-central-standalone.jar** file to this directory.
2. In a terminal window, navigate to the directory that contains the standalone JAR file.
3. Create the **application-users.properties** file. Include an administrative user and if this Business Central instance will be a Process Automation Manager controller for Process Server, include a Process Automation Manager controller user, for example:

```
rhpmAdmin=password1
controllerUser=controllerUser1234
```

4. Create the **application-roles.properties** file to assign roles to the users that you included in the **application-users.properties** file, for example:

```
rhpmAdmin=admin
controllerUser=kie-server
```

5. Create the **application-config.yaml** configuration file with the following contents, where **<APPLICATION_USERS>** is the path to the **application-users.properties** file and **<APPLICATION_ROLES>** is the path to the **application-roles.properties** file:

```
thorntail:
  security:
    security-domains:
      other:
        classic-authentication:
          login-modules:
            myloginmodule:
              code: org.kie.security.jaas.KieLoginModule
              flag: optional
              module: deployment.business-central-webapp.war
    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. Enter the following command:

```
java -jar rhpam-7.5.1-business-central-standalone.jar -s application-config.yaml
```

In addition, you can set any properties supported by Business Central by including the **-D<property>=<value>** parameter in this command, for example:

```
java -jar rhpam-7.5.1-business-central-standalone.jar -s application-config.yaml -
-D<property>=<value> -D<property>=<value>
```

See [Section 11.1, “Business Central system properties”](#) for more information.

11.1. BUSINESS CENTRAL SYSTEM PROPERTIES

The Business Central system properties listed in this section are passed to **standalone*.xml** files. To install standalone Business Central, you can use the listed properties in the following command:

```
java -jar rhpam-7.5.1-business-central-standalone.jar -s application-config.yaml -D<property>=
<value> -D<property>=<value>
```

In this command, **<property>** is a property from list and **<value>** is a value that you assign to that property.

Git directory

Use the following properties to set the location and name for the Business Central Git directory:

- **org.uberfire.nio.git.dir**: Location of the Business Central Git directory.
- **org.uberfire.nio.git.dirname**: Name of the Business Central Git directory. Default value: **.niogit**.
- **org.uberfire.nio.git.ketch**: Enables or disables Git ketch.
- **org.uberfire.nio.git.hooks**: Location of the Git hooks directory.

Git over HTTP

Use the following properties to configure access to the Git repository over HTTP:

- **org.uberfire.nio.git.proxy.ssh.over.http**: Specifies whether SSH should use an HTTP proxy. Default value: **false**.

- **http.proxyHost**: Defines the host name of the HTTP proxy. Default value: **null**.
- **http.proxyPort**: Defines the host port (integer value) of the HTTP proxy. Default value: **null**.
- **http.proxyUser**: Defines the user name of the HTTP proxy.
- **http.proxyPassword**: Defines the user password of the HTTP proxy.
- **org.uberfire.nio.git.http.enabled**: Enables or disables the HTTP daemon. Default value: **true**.
- **org.uberfire.nio.git.http.host**: If the HTTP daemon is enabled, it uses this property as the host identifier. This is an informative property that is used to display how to access the Git repository over HTTP. The HTTP still relies on the servlet container. Default value: **localhost**.
- **org.uberfire.nio.git.http.hostname**: If the HTTP daemon is enabled, it uses this property as the host name identifier. This is an informative property that is used to display how to access the Git repository over HTTP. The HTTP still relies on the servlet container. Default value: **localhost**.
- **org.uberfire.nio.git.http.port**: If the HTTP daemon is enabled, it uses this property as the port number. This is an informative property that is used to display how to access the Git repository over HTTP. The HTTP still relies on the servlet container. Default value: **8080**.

Git over HTTPS

Use the following properties to configure access to the Git repository over HTTPS:

- **org.uberfire.nio.git.proxy.ssh.over.https**: Specifies whether SSH uses an HTTPS proxy. Default value: **false**.
- **https.proxyHost**: Defines the host name of the HTTPS proxy. Default value: **null**.
- **https.proxyPort**: Defines the host port (integer value) of the HTTPS proxy. Default value: **null**.
- **https.proxyUser**: Defines the user name of the HTTPS proxy.
- **https.proxyPassword**: Defines the user password of the HTTPS proxy.
- **user.dir**: Location of the user directory.
- **org.uberfire.nio.git.https.enabled**: Enables or disables the HTTPS daemon. Default value: **false**.
- **org.uberfire.nio.git.https.host**: If the HTTPS daemon is enabled, it uses this property as the host identifier. This is an informative property that is used to display how to access the Git repository over HTTPS. The HTTPS still relies on the servlet container. Default value: **localhost**.
- **org.uberfire.nio.git.https.hostname**: If the HTTPS daemon is enabled, it uses this property as the host name identifier. This is an informative property that is used to display how to access the Git repository over HTTPS. The HTTPS still relies on the servlet container. Default value: **localhost**.
- **org.uberfire.nio.git.https.port**: If the HTTPS daemon is enabled, it uses this property as the port number. This is an informative property that is used to display how to access the Git repository over HTTPS. The HTTPS still relies on the servlet container. Default value: **8080**.

JGit

- **org.uberfire.nio.jgit.cache.instances:** Defines the JGit cache size.
- **org.uberfire.nio.jgit.cache.overflow.cleanup.size:** Defines the JGit cache overflow cleanup size.
- **org.uberfire.nio.jgit.remove.eldest.iterations:** Enables or disables whether to remove eldest JGit iterations.
- **org.uberfire.nio.jgit.cache.evict.threshold.duration:** Defines the JGit evict threshold duration.
- **org.uberfire.nio.jgit.cache.evict.threshold.time.unit:** Defines the JGit evict threshold time unit.

Git daemon

Use the following properties to enable and configure the Git daemon:

- **org.uberfire.nio.git.daemon.enabled:** Enables or disables the Git daemon. Default value: **true**.
- **org.uberfire.nio.git.daemon.host:** If the Git daemon is enabled, it uses this property as the local host identifier. Default value: **localhost**.
- **org.uberfire.nio.git.daemon.hostname:** If the Git daemon is enabled, it uses this property as the local host name identifier. Default value: **localhost**.
- **org.uberfire.nio.git.daemon.port:** If the Git daemon is enabled, it uses this property as the port number. Default value: **9418**.
- **org.uberfire.nio.git.http.sslVerify:** Enables or disables SSL certificate checking for Git repositories. Default value: **true**.



NOTE

If the default or assigned port is already in use, a new port is automatically selected. Ensure that the ports are available and check the log for more information.

Git SSH

Use the following properties to enable and configure the Git SSH daemon:

- **org.uberfire.nio.git.ssh.enabled:** Enables or disables the SSH daemon. Default value: **true**.
- **org.uberfire.nio.git.ssh.host:** If the SSH daemon is enabled, it uses this property as the local host identifier. Default value: **localhost**.
- **org.uberfire.nio.git.ssh.hostname:** If the SSH daemon is enabled, it uses this property as the local host name identifier. Default value: **localhost**.
- **org.uberfire.nio.git.ssh.port:** If the SSH daemon is enabled, it uses this property as the port number. Default value: **8001**.

**NOTE**

If the default or assigned port is already in use, a new port is automatically selected. Ensure that the ports are available and check the log for more information.

- **org.uberfire.nio.git.ssh.cert.dir**: Location of the **.security** directory where local certificates are stored. Default value: Working directory.
- **org.uberfire.nio.git.ssh.idle.timeout**: Sets the SSH idle timeout.
- **org.uberfire.nio.git.ssh.passphrase**: Pass phrase used to access the public key store of your operating system when cloning git repositories with SCP style URLs. Example: **git@github.com:user/repository.git**
- **org.uberfire.nio.git.ssh.algorithm**: Algorithm used by SSH. Default value: **RSA**.
- **org.uberfire.nio.git.gc.limit**: Sets the GC limit.
- **org.uberfire.nio.git.ssh.ciphers**: A comma-separated string of ciphers. The available ciphers are **aes128-ctr**, **aes192-ctr**, **aes256-ctr**, **arcfour128**, **arcfour256**, **aes192-cbc**, **aes256-cbc**. If the property is not used, all available ciphers are loaded.
- **org.uberfire.nio.git.ssh.macs**: A comma-separated string of message authentication codes (MACs). The available MACs are **hmac-md5**, **hmac-md5-96**, **hmac-sha1**, **hmac-sha1-96**, **hmac-sha2-256**, **hmac-sha2-512**. If the property is not used, all available MACs are loaded.

**NOTE**

If you plan to use RSA or any algorithm other than DSA, make sure you set up your application server to use the Bouncy Castle JCE library.

Process Server nodes and Process Automation Manager controller

Use the following properties to configure the connections with the Process Server nodes from the Process Automation Manager controller:

- **org.kie.server.controller**: The URL is used to connect to the Process Automation Manager controller. For example, **ws://localhost:8080/business-central/websocket/controller**.
- **org.kie.server.user**: User name used to connect to the Process Server nodes from the Process Automation Manager controller. This property is only required when using this Business Central installation as a Process Automation Manager controller.
- **org.kie.server.pwd**: Password used to connect to the Process Server nodes from the Process Automation Manager controller. This property is only required when using this Business Central installation as a Process Automation Manager controller.

Maven and miscellaneous

Use the following properties to configure Maven and other miscellaneous functions:

- **kie.maven.offline.force**: Forces Maven to behave as if offline. If true, disables online dependency resolution. Default value: **false**.



NOTE

Use this property for Business Central only. If you share a runtime environment with any other component, isolate the configuration and apply it only to Business Central.

- **org.uberfire.gzip.enable**: Enables or disables Gzip compression on the **GzipFilter** compression filter. Default value: **true**.
- **org.kie.workbench.profile**: Selects the Business Central profile. Possible values are **FULL** or **PLANNER_AND_RULES**. A prefix **FULL_** sets the profile and hides the profile preferences from the administrator preferences. Default value: **FULL**
- **org.appformer.m2repo.url**: Business Central uses the default location of the Maven repository when looking for dependencies. It directs to the Maven repository inside Business Central, for example, **http://localhost:8080/business-central/maven2**. Set this property before starting Business Central. Default value: File path to the inner **m2** repository.
- **appformer.ssh.keystore**: Defines the custom SSH keystore to be used with Business Central by specifying a class name. If the property is not available, the default SSH keystore is used.
- **appformer.ssh.keys.storage.folder**: When using the default SSH keystore, this property defines the storage folder for the user's SSH public keys. If the property is not available, the keys are stored in the Business Central **.security** folder.
- **appformer.experimental.features**: Enables the experimental features framework. Default value: **false**.
- **org.kie.demo**: Enables an external clone of a demo application from GitHub.
- **org.uberfire.metadata.index.dir**: Place where the Lucene **.index** directory is stored. Default value: Working directory.
- **org.uberfire ldap.regex.role_mapper**: Regex pattern used to map LDAP principal names to the application role name. Note that the variable **role** must be a part of the pattern as the application role name substitutes the variable **role** when matching a principle value and role name.
- **org.uberfire.sys.repo.monitor.disabled**: Disables the configuration monitor. Do not disable unless you are sure. Default value: **false**.
- **org.uberfire.secure.key**: Password used by password encryption. Default value: **org.uberfire.admin**.
- **org.uberfire.secure.alg**: Crypto algorithm used by password encryption. Default value: **PBEWithMD5AndDES**.
- **org.uberfire.domain**: Security-domain name used by uberfire. Default value: **ApplicationRealm**.
- **org.guvnor.m2repo.dir**: Place where the Maven repository folder is stored. Default value: **<working-directory>/repositories/kie**.
- **org.guvnor.project.gav.check.disabled**: Disables group ID, artifact ID, and version (GAV) checks. Default value: **false**.

- **org.kie.build.disable-project-explorer**: Disables automatic build of a selected project in Project Explorer. Default value: **false**.
- **org.kie.verification.disable-dtable-realtime-verification**: Disables the real-time validation and verification of decision tables. Default value: **false**.

Process Automation Manager controller

Use the following properties to configure how to connect to the Process Automation Manager controller:

- **org.kie.workbench.controller**: The URL used to connect to the Process Automation Manager controller, for example, **ws://localhost:8080/kie-server-controller/websocket/controller**.
- **org.kie.workbench.controller.user**: The Process Automation Manager controller user. Default value: **kieserver**.
- **org.kie.workbench.controller.pwd**: The Process Automation Manager controller password. Default value: **kieserver1!**.
- **org.kie.workbench.controller.token**: The token string used to connect to the Process Automation Manager controller.

Java Cryptography Extension KeyStore (JCEKS)

Use the following properties to configure JCEKS:

- **kie.keystore.keyStoreURL**: The URL used to load a Java Cryptography Extension KeyStore (JCEKS). For example, **file:///home/kie/keystores/keystore.jceks**.
- **kie.keystore.keyStorePwd**: The password used for the JCEKS.
- **kie.keystore.key.ctrl.alias**: The alias of the key for the default REST Process Automation Manager controller.
- **kie.keystore.key.ctrl.pwd**: The password of the alias for the default REST Process Automation Manager controller.

Rendering

Use the following properties to switch between Business Central and Process Server rendered forms:

- **org.jbpm.wb.forms.renderer.ext**: Switches the form rendering between Business Central and Process Server. By default, the form rendering is performed by Business Central. Default value: **false**.
- **org.jbpm.wb.forms.renderer.name**: Enables you to switch between Business Central and Process Server rendered forms. Default value: **workbench**.

CHAPTER 12. MAVEN SETTINGS AND REPOSITORIES FOR RED HAT PROCESS AUTOMATION MANAGER

When you create a Red Hat Process Automation Manager project, Business Central uses the Maven repositories that are configured for Business Central. You can use the Maven global or user settings to direct all Red Hat Process Automation Manager projects to retrieve dependencies from the public Red Hat Process Automation Manager repository by modifying the Maven project object model (POM) file (**pom.xml**). You can also configure Business Central and Process Server to use an external Maven repository or prepare a Maven mirror for offline use.

For more information about Red Hat Process Automation Manager packaging and deployment options, see [Packaging and deploying a Red Hat Process Automation Manager project](#) .

12.1. ADDING MAVEN DEPENDENCIES FOR RED HAT PROCESS AUTOMATION MANAGER

To use the correct Maven dependencies in your Red Hat Process Automation Manager project, add the Red Hat Business Automation bill of materials (BOM) files to the project's **pom.xml** file. The Red Hat Business Automation BOM applies to both Red Hat Decision Manager and Red Hat Process Automation Manager. When you add the BOM files, the correct versions of transitive dependencies from the provided Maven repositories are included in the project.

For more information about the Red Hat Business Automation BOM, see [What is the mapping between Red Hat Process Automation Manager and the Maven library version?](#).

Procedure

1. Declare the Red Hat Business Automation BOM in the **pom.xml** file:

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>com.redhat.ba</groupId>
      <artifactId>ba-platform-bom</artifactId>
      <version>7.5.1.redhat-00001</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
<dependencies>
  <!-- Your dependencies -->
</dependencies>
```

2. Declare dependencies required for your project in the **<dependencies>** tag. After you import the product BOM into your project, the versions of the user-facing product dependencies are defined so you do not need to specify the **<version>** sub-element of these **<dependency>** elements. However, you must use the **<dependency>** element to declare dependencies which you want to use in your project.
3. For standalone projects that are not authored in Business Central, specify all dependencies required for your projects. In projects that you author in Business Central, the basic decision engine and process engine dependencies are provided automatically by Business Central.
 - For a basic Red Hat Process Automation Manager project, declare the following

- For a basic Red Hat Process Automation Manager project, declare the following dependencies, depending on the features that you want to use:

Embedded process engine dependencies

```

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

<!-- Core dependencies for process engine -->
<dependency>
  <groupId>org.jbpm</groupId>
  <artifactId>jbpm-flow</artifactId>
</dependency>

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

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

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

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

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

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

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

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

```

```
</dependency>
```

```
<!-- Logging dependency. You can use any logging framework compatible with slf4j. -->
```

```
<dependency>
```

```
  <groupId>ch.qos.logback</groupId>
```

```
  <artifactId>logback-classic</artifactId>
```

```
  <version>${logback.version}</version>
```

```
</dependency>
```

- For a Red Hat Process Automation Manager project that uses CDI, you typically declare the following dependencies:

CDI-enabled process engine dependencies

```
<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>
```

- For a basic Red Hat Process Automation Manager project, declare the following dependencies:

Embedded decision engine dependencies

```
<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>

```

- To use the Process Server, declare the following dependencies:

Client application Process Server dependencies

```

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

```

- To create a remote client for Red Hat Process Automation Manager, declare the following dependency:

Client dependency

```

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

```

- When creating a JAR file that includes assets, such as rules and process definitions, specify the packaging type for your Maven project as **kjar** and use **org.kie:kie-maven-plugin** to process the **kjar** packaging type located under the **<project>** element. In the following example, **`\${kie.version}** is the Maven library version listed in [What is the mapping between Red Hat Process Automation Manager and the Maven library version?](#):

```

<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>

```

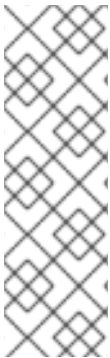
12.2. CONFIGURING AN EXTERNAL MAVEN REPOSITORY FOR BUSINESS CENTRAL AND PROCESS SERVER

You can configure Business Central and Process Server to use an external Maven repository, such as Nexus or Artifactory, instead of the built-in repository. This enables Business Central and Process Server to access and download artifacts that are maintained in the external Maven repository.



IMPORTANT

"Artifacts in the repository do not receive automated security patches as Maven requires that artifacts be immutable. As a result, artifacts that are missing patches for known security flaws will remain in the repository to avoid breaking builds that depend on them. Patched artifacts have their version number incremented". For more information, see [JBoss Enterprise Maven Repository](#).



NOTE

For information about configuring an external Maven repository for an authoring environment on Red Hat OpenShift Container Platform, see the following documents:

- [Deploying a Red Hat Process Automation Manager authoring environment on Red Hat OpenShift Container Platform](#)
- [Deploying a Red Hat Process Automation Manager environment on Red Hat OpenShift Container Platform using Operators](#)

Prerequisites

- Business Central and Process Server are installed. For installation options, see [Planning a Red Hat Process Automation Manager installation](#).

Procedure

1. Create a Maven **settings.xml** file with connection and access details for your external repository. For details about the **settings.xml** file, see the Maven [Settings Reference](#).
2. Save the file in a known location, for example, **/opt/custom-config/settings.xml**.
3. In your Red Hat Process Automation Manager installation directory, navigate to the **standalone-full.xml** file. For example, if you use a Red Hat JBoss EAP installation for Red Hat Process Automation Manager, go to **\$EAP_HOME/standalone/configuration/standalone-full.xml**.
4. Open **standalone-full.xml** and under the **<system-properties>** tag, set the **kie.maven.settings.custom** property to the full path name of the **settings.xml** file. For example:

```
<property name="kie.maven.settings.custom" value="/opt/custom-config/settings.xml"/>
```

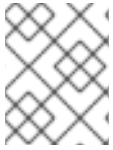
5. Start or restart Business Central and Process Server.

Next steps

For each Business Central project that you want to export or push as a KJAR artifact to the external Maven repository, you must add the repository information in the project **pom.xml** file. For instructions, see [Packaging and deploying a Red Hat Process Automation Manager project](#).

12.3. PREPARING A MAVEN MIRROR REPOSITORY FOR OFFLINE USE

If your Red Hat Process Automation Manager deployment does not have outgoing access to the public Internet, you must prepare a Maven repository with a mirror of all the necessary artifacts and make this repository available to your environment.



NOTE

You do not need to complete this procedure if your Red Hat Process Automation Manager deployment is connected to the Internet.

Prerequisites

- A computer that has outgoing access to the public Internet is available.

Procedure

1. On the computer that has an outgoing connection to the public Internet, complete the following steps:
 - a. Download the latest version of the [Offliner tool](#).
 - b. Download the **rhpmam-7.5.1-offliner.txt** product deliverable file from the [Software Downloads](#) page of the Red Hat Customer Portal.
 - c. Enter the following command to use the Offliner tool to download the required artifacts:

```
java -jar offliner-<version>.jar -r https://maven.repository.redhat.com/ga/ -r  
https://repo1.maven.org/maven2/ -d /home/user/temp rhpmam-7.5.1-offliner.txt
```

Replace **/home/user/temp** with an empty temporary directory and **<version>** with the version of the Offliner tool that you downloaded. The download can take a significant amount of time.

2. If you developed services outside Business Central and they have additional dependencies, add the dependencies to the mirror repository. If you developed the services as Maven projects, you can use the following steps to prepare these dependencies automatically. Complete the steps on the computer that has an outgoing connection to the public Internet.
 - a. Create a backup of the local Maven cache directory (**~/.m2/repository**) and then clear the directory.
 - b. Build the source of your projects using the **mvn clean install** command.
 - c. For every project, enter the following command to ensure that Maven downloads all runtime dependencies for all the artifacts generated by the project:

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

Replace **/path/to/project/pom.xml** with the correct path to the **pom.xml** file of the project.

- d. Copy the contents of the local Maven cache directory (**~/.m2/repository**) to the temporary directory that you are using.

3. Copy the contents of the temporary directory to a directory on the computer on which you deployed Red Hat Process Automation Manager. This directory becomes the offline Maven mirror repository.
4. Create and configure a **settings.xml** file for your Red Hat Process Automation Manager deployment, according to instructions in [Section 12.2, "Configuring an external Maven repository for Business Central and Process Server"](#).
5. Make the following changes in the **settings.xml** file:
 - Under the **<profile>** tag, if a **<repositories>** or **<pluginRepositories>** tag is absent, add the tags as necessary.
 - Under **<repositories>** add the following sequence:

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

Replace **/path/to/repo** with the full path to the local Maven mirror repository directory.

- Under **<pluginRepositories>** add the following sequence:

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

Replace **/path/to/repo** with the full path to the local Maven mirror repository directory.

6. Set the **kie.maven.offline.force** property for Business Central to **true**. For instructions about setting properties for Business Central, see [Section 11.1, "Business Central system properties"](#).

CHAPTER 13. IMPORTING PROJECTS FROM GIT REPOSITORIES

Git is a distributed version control system. It implements revisions as commit objects. When you save your changes to a repository, a new commit object in the Git repository is created.

Business Central uses Git to store project data, including assets such as rules and processes. When you create a project in Business Central, it is added to a Git repository that is embedded in Business Central. If you have projects in other Git repositories, you can import those projects into the Business Central Git repository through Business Central spaces.

Prerequisites

- Red Hat Process Automation Manager projects exist in an external Git repository.
- You have the credentials required for read access to that external Git repository.

Procedure

1. In Business Central, click **Menu → Design → Projects**.
2. Select or create the space into which you want to import the projects. The default space is **mySpace**.
3. Click the three vertical dots on the right side of the screen and select **Import Project**.
4. In the **Import Project** window, enter the URL and credentials for the Git repository that contains the projects that you want to import and click **Import**. The projects are added to the Business Central Git repository and are available from the current space.

CHAPTER 14. CUSTOMIZING THE BRANDING OF BUSINESS CENTRAL

You can customize the branding of the Business Central login page and application header by replacing the images with your own.

14.1. CUSTOMIZING THE BUSINESS CENTRAL LOGIN PAGE

You can customize the company logo and the project logo on the Business Central login page.

Procedure

1. Start Red Hat JBoss EAP and open Business Central in a web browser.
2. Copy a PNG format image to the **`EAP_HOME/standalone/deployments/business-central.war/img/`** directory in your Red Hat Process Automation Manager installation.
3. In the **`EAP_HOME/standalone/deployments/business-central.war/img/`** directory, either move or rename the existing **`login-screen-logo.png`** file.
4. Rename your PNG file **`login-screen-logo.png`**.
5. To change the project logo that appears above the **User name** and **Password** fields, replace the default image **`RHPAM_Logo.svg`** with a new SVG file.
6. Force a full reload of the login page, bypassing the cache, to view the changes. For example, in most Linux and Windows web browsers, press Ctrl+F5.

14.2. CUSTOMIZING BUSINESS CENTRAL APPLICATION HEADER

You can customize the Business Central application header.

Procedure

1. Start Red Hat JBoss EAP, open Business Central in a web browser, and log in with your user credentials.
2. Copy your new application header image in the SVG format to the **`EAP_HOME/standalone/deployments/business-central.war/banner/`** directory in your Red Hat Process Automation Manager installation.
3. Open the **`EAP_HOME/standalone/deployments/business-central.war/banner/banner.html`** file in a text editor.
4. Replace **`logo.svg`** in the **``** tag with the file name of your new image: **`admin-and-config/`**

```

```
5. Force a full reload of the login page, bypassing the cache, to view the changes. For example, in most Linux and Windows web browsers, press Ctrl+F5.

CHAPTER 15. INTEGRATING LDAP AND SSL

With Red Hat Process Automation Manager you can integrate LDAP and SSL through RH-SSO. For more information, see the [Red Hat Single Sign-On Server Administration Guide](#) .

CHAPTER 16. ADDITIONAL RESOURCES

- *Installing and configuring Red Hat Process Automation Manager in a Red Hat JBoss EAP clustered environment*

CHAPTER 17. NEXT STEPS

- *Integrating Red Hat Process Automation Manager with Red Hat Single Sign-On*
- *Getting started with business processes*

APPENDIX A. VERSIONING INFORMATION

Documentation last updated on Wednesday, July 15, 2020.