Red Hat Decision Manager 7.13

Installing and configuring Red Hat Decision Manager

Last Updated: 2024-03-14

Red Hat Decision Manager 7.13 Installing and configuring Red Hat Decision Manager

Legal Notice

Copyright © 2024 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

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. Starting with this release, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files. There are no Decision Manager artifacts. The Red Hat Decision Manager subscription, support entitlements, and fees remain the same. Red Hat Decision Manager subscribers will continue to receive full support for the decision management and optimization capabilities of Red Hat Decision Manager. The business process management (BPM) capabilities of Red Hat Process Automation Manager are exclusive to Red Hat Process Automation Manager subscribers. They are available for use by Red Hat Decision Manager subscribers but with

development support services only. Red Hat Decision Manager subscribers can upgrade to a full Red Hat Process Automation Manager subscription at any time to receive full support for BPM features.

Table of Contents

PREFACE 7

MAKING OPEN SOURCE MORE INCLUSIVE 8

PART I. PLANNING A RED HAT PROCESS AUTOMATION INSTALLATION 9

CHAPTER 1. RED HAT DECISION MANAGER VERSIONING 10

CHAPTER 2. RED HAT DECISION MANAGER COMPONENTS 12

CHAPTER 3. RED HAT DECISION MANAGER INSTALLATION PLATFORMS 13
3.1. RED HAT JBOSS EAP 13
3.2. RED HAT JBOSS WEB SERVER 13
3.3. RED HAT OPENSHIFT CONTAINER PLATFORM 13
3.4. ORACLE WEBLOGIC SERVER 14
3.5. IBM WEBSPHERE APPLICATION SERVER 14

CHAPTER 4. RED HAT DECISION MANAGER SUPPORTED REPOSITORIES 15
4.1. GIT REPOSITORIES 15
4.2. APACHE MAVEN 15

CHAPTER 5. INSTALLATION ENVIRONMENT OPTIONS FOR RED HAT DECISION MANAGER 16

CHAPTER 6. RED HAT DECISION MANAGER ROLES AND USERS 17

CHAPTER 7. RED HAT DECISION MANAGER REFERENCE IMPLEMENTATIONS 18

CHAPTER 8. INTEGRATING WITH RED HAT DECISION MANAGER 19
8.1. RED HAT DECISION MANAGER SPRING BOOT BUSINESS APPLICATIONS 19
8.2. INTEGRATING RED HAT FUSE 19
8.3. INTEGRATING RED HAT SINGLE SIGN-ON 19

CHAPTER 9. ADDITIONAL RESOURCES 20

PART II. INSTALLING AND CONFIGURING RED HAT DECISION MANAGER ON RED HAT JBOSS EAP 7.4 21

CHAPTER 10. ABOUT RED HAT DECISION MANAGER 22

CHAPTER 11. RED HAT DECISION MANAGER ROLES AND USERS 24

CHAPTER 12. DOWNLOADING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION FILES 25

CHAPTER 13. USING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLER 26
13.1. USING THE INSTALLER IN INTERACTIVE MODE 26
13.2. USING THE INSTALLER IN CLI MODE 28

CHAPTER 14. INSTALLING RED HAT PROCESS AUTOMATION MANAGER FROM ZIP FILES 31
14.1. INSTALLING BUSINESS CENTRAL FROM THE ZIP FILE 31
14.2. INSTALLING KIE SERVER FROM THE ZIP FILE 31
14.3. CREATING USERS 32
14.4. CONFIGURING KIE SERVER TO CONNECT TO BUSINESS CENTRAL 35
14.5. THREAD EFFICIENCY 37

CHAPTER 15. SECURING PASSWORDS WITH A KEYSTORE 39

CHAPTER 16. CONFIGURING SSH TO USE RSA 41

CHAPTER 17. VERIFYING THE RED HAT DECISION MANAGER INSTALLATION 42

CHAPTER 18. RUNNING RED HAT PROCESS AUTOMATION MANAGER 43

CHAPTER 19. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 45
19.1. INSTALLING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 45
19.1.1. Creating a headless Process Automation Manager controller user 46
19.1.2. Configuring KIE Server and the headless Process Automation Manager controller 47
19.2. RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 48

CHAPTER 20. CONFIGURING AND RUNNING STANDALONE BUSINESS CENTRAL 50

CHAPTER 21. MAVEN SETTINGS AND REPOSITORIES FOR RED HAT DECISION MANAGER 52
21.1. ADDING MAVEN DEPENDENCIES FOR RED HAT DECISION MANAGER 52
21.2. CONFIGURING AN EXTERNAL MAVEN REPOSITORY FOR BUSINESS CENTRAL AND KIE SERVER 54
21.3. PREPARING A MAVEN MIRROR REPOSITORY FOR OFFLINE USE 55

CHAPTER 22. IMPORTING PROJECTS FROM GIT REPOSITORIES 58

CHAPTER 23. CUSTOMIZING THE BRANDING OF BUSINESS CENTRAL 59
23.1. CUSTOMIZING THE BUSINESS CENTRAL LOGIN PAGE 59
23.2. CUSTOMIZING BUSINESS CENTRAL APPLICATION HEADER 59

CHAPTER 24. INTEGRATING LDAP AND SSL 60

APPENDIX A. BUSINESS CENTRAL SYSTEM PROPERTIES 61

CHAPTER 25. ADDITIONAL RESOURCES 67

CHAPTER 26. NEXT STEPS 68

PART III. INSTALLING AND CONFIGURING RED HAT PROCESS AUTOMATION MANAGER IN A RED HAT JBOSS EAP CLUSTERED ENVIRONMENT 69

CHAPTER 27. RED HAT DECISION MANAGER CLUSTERS 70

CHAPTER 28. RED HAT PROCESS AUTOMATION MANAGER CLUSTERS IN A DEVELOPMENT (AUTHORING) ENVIRONMENT 71
28.1. INSTALLING AND CONFIGURING RED HAT DATA GRID 71
28.2. DOWNLOADING AND CONFIGURING AMQ BROKER 72
28.3. CONFIGURING AN NFS VERSION 4 SERVER 73
28.4. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.4 AND RED HAT PROCESS AUTOMATION MANAGER 74
28.5. CONFIGURING AND RUNNING BUSINESS CENTRAL IN A CLUSTER 75
28.6. TESTING YOUR HIGH AVAILABILITY (HA) ON-PREMISE INFRASTRUCTURE 77
28.7. VERIFYING THE RED HAT PROCESS AUTOMATION MANAGER CLUSTER 79

CHAPTER 29. KIE SERVER CLUSTERS IN A RUNTIME ENVIRONMENT 81
29.1. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.4 AND KIE SERVER 81
29.2. CLUSTERING KIE SERVERS WITH THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 83

CHAPTER 30. ADDITIONAL RESOURCES 85

PART IV. INSTALLING AND CONFIGURING RED HAT PROCESS AUTOMATION MANAGER ON RED HAT JBOSS WEB SERVER 86

CHAPTER 31. ABOUT RED HAT DECISION MANAGER 87

31.1. RED HAT DECISION MANAGER COMPONENTS 88
31.2. RED HAT DECISION MANAGER ROLES AND USERS 88

CHAPTER 32. DOWNLOADING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION FILES .. 90

CHAPTER 33. USING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLER 91
33.1. USING THE INSTALLER IN INTERACTIVE MODE 91
33.2. USING THE INSTALLER IN CLI MODE 93

CHAPTER 34. KIE SERVER ZIP FILE INSTALLATION AND CONFIGURATION 96
34.1. INSTALLING KIE SERVER FROM ZIP FILES 96

CHAPTER 35. SECURING PASSWORDS WITH A KEYSTORE 98

CHAPTER 36. VERIFYING THE KIE SERVER INSTALLATION 101

CHAPTER 37. DOWNLOADING AND INSTALLING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 102

CHAPTER 38. CONFIGURING AND RUNNING STANDALONE BUSINESS CENTRAL 105

CHAPTER 39. MAVEN SETTINGS AND REPOSITORIES FOR RED HAT DECISION MANAGER 107
39.1. CONFIGURING MAVEN USING THE PROJECT CONFIGURATION FILE (POM.XML) 107
39.2. MODIFYING THE MAVEN SETTINGS FILE 107
39.3. ADDING MAVEN DEPENDENCIES FOR RED HAT DECISION MANAGER 108
39.4. PREPARING A MAVEN MIRROR REPOSITORY FOR OFFLINE USE 110

CHAPTER 40. IMPORTING PROJECTS FROM GIT REPOSITORIES 113

CHAPTER 41. INTEGRATING LDAP AND SSL 114

APPENDIX B. BUSINESS CENTRAL SYSTEM PROPERTIES 115

PART V. INSTALLING AND CONFIGURING KIE SERVER ON IBM WEBSPHERE APPLICATION SERVER . 121

CHAPTER 42. KIE SERVER 122

CHAPTER 43. IBM WEBSPHERE APPLICATION SERVER 123

CHAPTER 44. INSTALLING AND RUNNING IBM WEBSPHERE APPLICATION SERVER 124

CHAPTER 45. CONFIGURING IBM WEBSPHERE APPLICATION SERVER FOR KIE SERVER 126
45.1. ENABLING ADMINISTRATIVE SECURITY 126
45.2. CONFIGURING JAVA MESSAGE SERVICE (JMS) 127
45.2.1. Create a service bus and add IBM WebSphere Application Server 127
45.2.2. Create JMS connection factories 127
45.2.2.1. JMS connection factories for KIE Server 128
45.2.3. Create JMS queues 128
45.2.3.1. JMS queues for KIE Server 128
45.2.4. Create JMS activation specifications 129
45.2.4.1. JMS activation specifications for KIE Server 129
45.3. SETTING SYSTEM PROPERTIES IN IBM WEBSPHERE APPLICATION SERVER 130
45.4. STOPPING AND RESTARTING IBM WEBSPHERE APPLICATION SERVER 132

CHAPTER 46. INSTALLING AND RUNNING KIE SERVER WITH IBM WEBSPHERE APPLICATION SERVER ... 134
46.1. CREATING THE KIE SERVER GROUP AND ROLE 135
46.2. MAPPING THE KIE SERVER GROUP AND ROLE 135

46.3. CONFIGURING CLASS LOADING FOR KIE SERVER 136
46.4. VERIFYING THE INSTALLATION 136

CHAPTER 47. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER WITH IBM WEBSPHERE APPLICATION SERVER 137
47.1. CONFIGURING CLASS LOADING FOR THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 138
47.2. SETTING SYSTEM PROPERTIES FOR THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 138
47.3. MAPPING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER GROUP AND ROLE 140
47.4. VERIFYING THE INSTALLATION 140

CHAPTER 48. CONFIGURING AN EMBEDDED DECISION ENGINE IN IBM WEBSPHERE APPLICATION SERVER 142

CHAPTER 49. SECURING PASSWORDS WITH A KEYSTORE 145

CHAPTER 50. NEXT STEPS 147

PART VI. INSTALLING AND CONFIGURING KIE SERVER ON ORACLE WEBLOGIC SERVER 148

CHAPTER 51. KIE SERVER 149

CHAPTER 52. ORACLE WEBLOGIC SERVER 150

CHAPTER 53. INSTALLING AND RUNNING ORACLE WEBLOGIC SERVER 151

CHAPTER 54. CONFIGURING ORACLE WEBLOGIC SERVER FOR KIE SERVER 153
54.1. CONFIGURING THE KIE SERVER GROUP AND USERS 153
54.2. CONFIGURING JAVA MESSAGE SERVICE (JMS) 153
54.2.1. Create a JMS server 153
54.2.2. Create a JMS module 154
54.2.3. Create JMS connection factories 154
54.2.4. Create JMS queues 155
54.3. SETTING SYSTEM PROPERTIES IN ORACLE WEBLOGIC SERVER 156
54.4. STOPPING AND RESTARTING ORACLE WEBLOGIC SERVER 157

CHAPTER 55. INSTALLING KIE SERVER WITH ORACLE WEBLOGIC SERVER 159
55.1. VERIFYING THE KIE SERVER INSTALLATION ON ORACLE WEBLOGIC SERVER 159

CHAPTER 56. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER WITH ORACLE WEBLOGIC SERVER 161
56.1. SETTING SYSTEM PROPERTIES FOR THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER 162
56.2. VERIFYING THE INSTALLATION 163

CHAPTER 57. CONFIGURING AN EMBEDDED DECISION ENGINE IN ORACLE WEBLOGIC SERVER 165

CHAPTER 58. SECURING PASSWORDS WITH A KEYSTORE 168

CHAPTER 59. NEXT STEPS 170

PART VII. PATCHING AND UPGRADING RED HAT DECISION MANAGER 171

CHAPTER 60. RED HAT DECISION MANAGER VERSIONING 172

CHAPTER 61. ENABLING NOTIFICATIONS IN THE RED HAT CUSTOMER PORTAL 174

CHAPTER 62. APPLYING PATCH UPDATES AND MINOR RELEASE UPGRADES TO RED HAT PROCESS

AUTOMATION MANAGER 175

CHAPTER 63. ADDITIONAL RESOURCES 179

APPENDIX C. VERSIONING INFORMATION 180

APPENDIX D. CONTACT INFORMATION 181

PREFACE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. Starting with this release, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files. There are no Decision Manager artifacts. The Red Hat Decision Manager subscription, support entitlements, and fees remain the same. Red Hat Decision Manager subscribers will continue to receive full support for the decision management and optimization capabilities of Red Hat Decision Manager. The business process management (BPM) capabilities of Red Hat Process Automation Manager are exclusive to Red Hat Process Automation Manager subscribers. They are available for use by Red Hat Decision Manager subscribers but with development support services only. Red Hat Decision Manager subscribers can upgrade to a full Red Hat Process Automation Manager subscription at any time to receive full support for BPM features.

As a developer or system administrator, you can install Red Hat Decision Manager components on Red Hat JBoss EAP, in an Red Hat JBoss EAP clustered environment, on JBoss Web Server, and in other supported environments.

MAKING OPEN SOURCE MORE INCLUSIVE

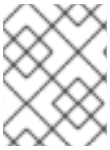
Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see *our CTO Chris Wright's message* .

PART I. PLANNING A RED HAT PROCESS AUTOMATION INSTALLATION

As a system administrator, you have several options for installing Red Hat Process Automation.

CHAPTER 1. RED HAT DECISION MANAGER VERSIONING

Red Hat Process Automation Manager versions are designated with a numerical **Major.Minor.Patch** format, such as 7.13.5. In this example, the major release is **7.x.x**, the minor release is **7.13.x**, and the patch release is 7.13.5. Major releases often require data migration, while minor release upgrades and patch updates are typically managed with update tools provided with the Red Hat Decision Manager release artifacts.



NOTE

Starting with release 7.13, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files.

The following are the general types of releases for Red Hat Decision Manager:

Major release migrations

Major releases of Red Hat Decision Manager include substantial enhancements, security updates, bug fixes, and possibly redesigned features and functions. Data migration is typically required when an application is moved from one major release to another major release, such as from Red Hat JBoss BRMS 6.4.x to Red Hat Decision Manager 7.0. Automated migration tools are often provided with new major versions of Red Hat Decision Manager to facilitate migration, but some manual effort is likely required for certain data and configurations. The supported migration paths are specified in product announcements and documentation.

For example migration instructions, see *Migrating from Red Hat JBoss BRMS 6.4 to Red Hat Decision Manager 7.0*.

Minor release upgrades

Minor releases of Red Hat Decision Manager include enhancements, security updates, and bug fixes. Data migration may be required when an application is moved from one minor release to another minor release, such as from Red Hat Decision Manager 7.5.x to 7.6. Automated update tools are often provided with both patch updates and new minor versions of Red Hat Decision Manager to facilitate updating certain components of Red Hat Decision Manager, such as Business Central, KIE Server, and the headless Process Automation Manager controller. Other Red Hat Decision Manager artifacts, such as the decision engine and standalone Business Central, are released as new artifacts with each minor release and you must reinstall them to apply the update.

Before you upgrade to a new minor release, apply the latest patch update to your current version of Red Hat Decision Manager to ensure that the minor release upgrade is successful.

Patch updates

Patch updates of Red Hat Decision Manager include the latest security updates and bug fixes. Scheduled patch updates contain all previously released patch updates for that minor version of the product, so you do not need to apply each patch update incrementally in order to apply the latest update. For example, you can update Red Hat Decision Manager 7.5.0 or 7.5.1 to Red Hat Decision Manager 7.5.2. However, for optimal Red Hat Decision Manager performance, apply product updates as they become available. Occasionally, Red Hat might release unscheduled patch updates outside the normal update cycle of the existing product. These may include security or other updates provided by Red Hat Global Support Services (GSS) to fix specific issues, and may not be cumulative updates.

Automated update tools are often provided with both patch updates and new minor versions of Red Hat Decision Manager to facilitate updating certain components of Red Hat Decision Manager, such as Business Central, KIE Server, and the headless Process Automation Manager controller. Other

Red Hat Decision Manager artifacts, such as the decision engine and standalone Business Central, are released as new artifacts with each minor release and you must reinstall them to apply the update.

To ensure optimal transition between releases and to keep your Red Hat Decision Manager distribution current with the latest enhancements and fixes, apply new product releases and updates to Red Hat Decision Manager as they become available in the Red Hat Customer Portal. Consider also enabling product notifications in the Red Hat Customer Portal.

CHAPTER 2. RED HAT DECISION MANAGER COMPONENTS

The product is made up of Business Central and KIE Server.

- Business Central is the graphical user interface where you create and manage business rules. You can install Business Central in a Red Hat JBoss EAP instance or on the Red Hat OpenShift Container Platform (OpenShift).
  Business Central is also available as a standalone JAR file. You can use the Business Central standalone JAR file to run Business Central without deploying it to an application server.

- KIE Server is the server where rules and other artifacts are executed. It is used to instantiate and execute rules and solve planning problems. You can install KIE Server in a Red Hat JBoss EAP instance, in a Red Hat JBoss EAP cluster, on OpenShift, in an Oracle WebLogic server instance, in an IBM WebSphere Application Server instance, or as a part of Spring Boot application.
  You can configure KIE Server to run in managed or unmanaged mode. If KIE Server is unmanaged, you must manually create and maintain KIE containers (deployment units). A KIE container is a specific version of a project. If KIE Server is managed, the Process Automation Manager controller manages the KIE 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 on Red Hat JBoss EAP, use the Execution Server page to create and maintain KIE containers. However, if you do not install Business Central, you can install the headless Process Automation Manager controller and use the REST API or the KIE Server Java Client API to interact with it.

- Red Hat build of OptaPlanner is integrated in Business Central and KIE Server. It is a lightweight, embeddable planning engine that optimizes planning problems. Red Hat build of OptaPlanner helps Java programmers solve planning problems efficiently, and it combines optimization heuristics and metaheuristics with efficient score calculations.

CHAPTER 3. RED HAT DECISION MANAGER INSTALLATION PLATFORMS

Every Red Hat Decision Manager release is certified on a variety of operating systems, JVMs, web browsers, and database combinations. Red Hat provides both production and development support for supported and tested configurations in both physical and virtual environments, according to your subscription agreement. See the following pages for information about supported configurations and version numbers:

- Red Hat Decision Manager 7 Supported Configurations
- Red Hat Decision Manager 7 Component Details

Red Hat Decision Manager 7.13 is available for the following application platforms:

- Red Hat JBoss EAP
- Red Hat JBoss Web Server
- Red Hat OpenShift Container Platform
- Oracle WebLogic Server
- IBM WebSphere Application Server

3.1. RED HAT JBOSS EAP

Red Hat JBoss Enterprise Application Platform (Red Hat JBoss EAP) is a certified implementation of the Java Enterprise Edition 7 (Java EE 7) 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.

You can install both Business Central and KIE Server in a single Red Hat JBoss EAP instance. However, in production environments you should install them in separate instances.

3.2. RED HAT JBOSS WEB SERVER

Red Hat JBoss Web Server is an enterprise-ready web server designed for medium and large applications, based on Tomcat. Red Hat JBoss Web Server provides organizations with a single deployment platform for Java Server Pages (JSP) and Java Servlet technologies, PHP, and CGI.

You can install KIE Server and the headless Process Automation Manager controller on Red Hat JBoss Web Server.

3.3. RED HAT OPENSHIFT CONTAINER PLATFORM

Red Hat OpenShift Container Platform (OpenShift) uses Kubernetes and provides an API to manage these services. OpenShift enables you to create and manage containers.

In this solution, components of Red Hat Decision Manager are deployed as separate OpenShift pods. You can scale each of the pods up and down individually, providing as few or as many containers as necessary for a particular component. You can use standard OpenShift methods to manage the pods and balance the load.

You can install both Business Central and KIE Server on OpenShift.

3.4. ORACLE WEBLOGIC SERVER

Oracle WebLogic Server is a Java EE application server that provides a standard set of APIs for creating distributed Java applications that can access a wide variety of services, such as databases, messaging services, and connections to external enterprise systems. Users access these applications using web browser clients or Java clients.

You can install KIE Server in an Oracle WebLogic Server instance and then use the headless Process Automation Manager controller and the REST API or the KIE Server Java Client API to interact with KIE Server. You can also use standalone Business Central to interact with KIE Server.

3.5. IBM WEBSPHERE APPLICATION SERVER

IBM WebSphere Application Server is a flexible and secure web application server that hosts Java-based web applications and provides Java EE-certified run time environments. IBM WebSphere 9.0 supports Java SE 8 and is fully compliant with Java EE 7 since version 8.5.5.6.

You can install KIE Server in an IBM WebSphere Application Server instance and then use the headless Process Automation Manager controller to interact with KIE Server. You can also use standalone Business Central to interact with KIE Server.

CHAPTER 4. RED HAT DECISION MANAGER SUPPORTED REPOSITORIES

You can use Git repositories and Apache Maven with Red Hat Decision Manager.

4.1. GIT REPOSITORIES

Git repositories are used internally within Business Central to store all processes, rules, and other artifacts that are created in the authoring environment. Git is a distributed version control system. It implements revisions as commit objects. When you commit your changes into a repository, a new commit object in the Git repository is created. When you create a project in Business Central, it is added to the Git repository that is connected to Business Central.

If you have projects in other Git repositories, you can import them into Business Central spaces. You can use Git hooks to synchronize the internal Git repositories with external repositories.

4.2. APACHE MAVEN

Apache Maven is a distributed build automation tool used in Java application development to build and manage software projects. You can use Maven to build, publish, and deploy your Red Hat Decision Manager projects. Maven provides the following benefits:

- The build process is easy and a uniform build system is implemented across projects.
- All of the required JAR files for a project are made available at compile time.
- A proper project structure is configured.
- Dependencies and versions are well managed.
- There is no need for additional build processing because Maven builds output into a number of predefined types, such as JAR and WAR.

Maven uses repositories to store Java libraries, plug-ins, and other build artifacts. These repositories can be local or remote. Red Hat Decision Manager maintains local and remote maven repositories that you can add to your project for accessing the rules, processes, events, and other project dependencies. When building projects and archetypes, Maven dynamically retrieves Java libraries and Maven plug-ins from local or remote repositories. Doing this promotes sharing and reuse of dependencies across projects.

CHAPTER 5. INSTALLATION ENVIRONMENT OPTIONS FOR RED HAT DECISION MANAGER

With Red Hat Process Automation Manager, you can set up a development environment to develop business applications, a runtime environment to run those applications to support decisions, or both.

- **Development environment**: Typically consists of one Business Central installation and at least one KIE Server installation. You can use Business Central to design decisions and other artifacts, and you can use KIE Server to execute and test the artifacts that you created.
- **Runtime environment**: Consists of one or more KIE Server instances with or without Business Central. Business Central has an embedded Process Automation Manager controller. If you install Business Central, use the **Menu → Deploy → Execution servers** page to create and maintain containers. If you want to automate KIE Server management without Business Central, you can use the headless Process Automation Manager controller.

You can also cluster both development and runtime environments. A clustered development or runtime environment consists of a unified group or cluster of two or more servers. The primary benefit of clustering Red Hat Process Automation Manager development environments is high availability and enhanced collaboration, while the primary benefit of clustering Red Hat Process Automation Manager runtime environments is high availability and load balancing. High availability decreases the chance of data loss when a single server fails. When a server fails, another server fills the gap by providing a copy of the data that was on the failed server. When the failed server comes online again, it resumes its place in the cluster.

NOTE

Clustering of the runtime environment is currently supported on Red Hat JBoss EAP 7.4 and Red Hat OpenShift Container Platform only.

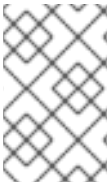
CHAPTER 6. RED HAT DECISION MANAGER ROLES AND USERS

To access Business Central or KIE Server, you must create users and assign them appropriate roles before the servers are started. You can create users and roles when you install Business Central or KIE Server.

If both Business Central and KIE Server are running on a single instance, a user who is authenticated for Business Central can also access KIE Server.

However, if Business Central and KIE Server are running on different instances, a user who is authenticated for Business Central must be authenticated separately to access KIE Server. For example, if a user who is authenticated on Business Central but not authenticated on KIE Server tries to view or manage process definitions in Business Central, a 401 error is logged in the log file and the **Invalid credentials to load data from remote server. Contact your system administrator.** message appears in Business Central.

This section describes Red Hat Decision Manager user roles.



NOTE

The **admin**, **analyst**, and **rest-all** roles are reserved for Business Central. The **kie-server** role is reserved for KIE Server. For this reason, the available roles can differ depending on whether Business Central, KIE 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 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 Decision Manager.
- **analyst**: Users with the **analyst** role have access to all high-level features. They can model 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.
- **rest-all**: Users with the **rest-all** role can access Business Central REST capabilities.
- **kie-server**: Users with the **kie-server** role can access KIE Server REST capabilities.

CHAPTER 7. RED HAT DECISION MANAGER REFERENCE IMPLEMENTATIONS

Red Hat Decision Manager provides reference implementations that you can use as starter applications. They are included in the **Red Hat Process Automation Manager 7.13.5 Reference Implementations** download, available on the Red Hat Decision Manager Software Downloads page in the Red Hat Customer Portal.

Employee Rostering reference implementation

The employee rostering reference implementation enables you to create an application that assigns employees to shifts on various positions in an organization. For example, you can use the application to distribute shifts in a hospital between nurses, guard duty shifts across a number of locations, or shifts on an assembly line between workers.

Vehicle route planning reference implementation

The vehicle route planning reference implementation enables you to create an application that solves a vehicle route planning problem with real-world maps, roads, and vehicles delivering goods to locations, each with a specific capacity. For more information, see the README file in the vehicle routing ZIP file, included in the reference implementation download.

School timetable reference implementation

The school timetable reference implementation enables you to build a REST application on Spring Boot that associates lessons with rooms and time slots and avoids conflicts by considering student and teacher constraints.

High available event-driven decisioning reference implementation

The high available event-driven decisioning reference implementation enables you to deploy Drools engine code that requires stateful processing, including rules developed with complex event processing, in an OpenShift environment. Doing this enables the decision engine to process complex event series with high availability.

CHAPTER 8. INTEGRATING WITH RED HAT DECISION MANAGER

You can integrate Red Hat Decision Manager with other products and components, such as Spring Boot, Red Hat Fuse, and Red Hat Single Sign-On.

8.1. RED HAT DECISION MANAGER SPRING BOOT BUSINESS APPLICATIONS

Spring Framework is a Java platform that provides comprehensive infrastructure support for developing Java applications. Spring Boot is a lightweight framework based on Spring Boot starters. Spring Boot starters are **pom.xml** files that contain a set of dependency descriptors that you can include in your Spring Boot project. Red Hat offers support and maintenance for Red Hat Decision Manager Spring Boot projects as part of Red Hat Runtimes.

Red Hat Decision Manager Spring Boot business applications are flexible, UI-agnostic logical groupings of individual services that provide certain business capabilities. Business applications are based on Spring Boot starters. They are usually deployed separately and can be versioned individually. A complete business application enables a domain to achieve specific business goals, for example, order management or accommodation management. After you create and configure your business application, you can deploy it to an existing service or to the cloud, through OpenShift.

8.2. INTEGRATING RED HAT FUSE

Fuse is a distributed, cloud-native integration platform that is part of an agile integration solution. Its distributed approach enables teams to deploy integrated services where required.

You can install Red Hat Fuse on the Apache Karaf container platform or Red Hat JBoss Enterprise Application Platform and then install and configure Red Hat Decision Manager in that container.

8.3. INTEGRATING RED HAT SINGLE SIGN-ON

Red Hat Single Sign-On (RH-SSO) is a single sign-on solution that you can use to secure your browser applications with your REST web services and Git access.

When you integrate Red Hat Decision Manager with RH-SSO, you create an SSO and identity management (IDM) environment for Red Hat Decision Manager. The session management feature of RH-SSO enables you to use a single authentication for different Red Hat Decision Manager environments on the internet.

Red Hat Decision Manager supports RH-SSO 7.5 on Red Hat JBoss EAP 7.4.

CHAPTER 9. ADDITIONAL RESOURCES

Installation

- *Installing and configuring Red Hat Decision Manager on Red Hat JBoss EAP 7.4*
- *Installing and configuring Red Hat Decision Manager in a Red Hat JBoss EAP clustered environment*
- *Installing and configuring Red Hat Decision Manager on Red Hat JBoss Web Server*
- *Installing and configuring KIE Server on IBM WebSphere Application Server*
- *Installing and configuring KIE Server on Oracle WebLogic Server*

Integration

- *Creating Red Hat Decision Manager business applications with Spring Boot*
- *Integrating Red Hat Fuse with Red Hat Decision Manager*
- *Integrating Red Hat Decision Manager with Red Hat Single Sign-On*

Red Hat build of OptaPlanner

- *Developing Solvers with Red Hat Decision Manager*

OpenShift

- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 4 using Operators*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 3 using templates*

PART II. INSTALLING AND CONFIGURING RED HAT DECISION MANAGER ON RED HAT JBOSS EAP 7.4

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. Starting with this release, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files. There are no Decision Manager artifacts. The Red Hat Decision Manager subscription, support entitlements, and fees remain the same. Red Hat Decision Manager subscribers will continue to receive full support for the decision management and optimization capabilities of Red Hat Decision Manager. The business process management (BPM) capabilities of Red Hat Process Automation Manager are exclusive to Red Hat Process Automation Manager subscribers. They are available for use by Red Hat Decision Manager subscribers but with development support services only. Red Hat Decision Manager subscribers can upgrade to a full Red Hat Process Automation Manager subscription at any time to receive full support for BPM features.

This document describes how to install Red Hat Process Automation Manager on a Red Hat JBoss Enterprise Application Platform 7.4 instance.

Prerequisites

- You have reviewed the information in *Planning a Red Hat Decision Manager installation* .
- You have installed the latest patch release of Red Hat JBoss Enterprise Application Platform 7.4. For information about installing Red Hat JBoss EAP, see the *Red Hat JBoss Enterprise Application Platform installation guide*.

CHAPTER 10. ABOUT RED HAT DECISION MANAGER

Red Hat Decision Manager is an open source decision management platform that combines business rules management, complex event processing, Decision Model & Notation (DMN) execution, and Red Hat build of OptaPlanner for solving planning problems. It automates business decisions and makes that logic available to the entire business.

Business assets such as rules, decision tables, and DMN models are organized in projects and stored in the Business Central repository. This ensures consistency, transparency, and the ability to audit across the business. Business users can modify business logic without requiring assistance from IT personnel.

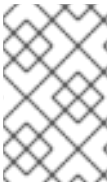
The product is made up of Business Central and KIE Server.

- KIE Server is the server where rules and other artifacts are executed. It is used to instantiate and execute rules and solve planning problems. KIE Server provides the runtime environment for business assets and accesses the data stored in the assets repository (knowledge store).
- Business Central is the graphical user interface where you create and manage business rules that KIE Server executes.

Business Central is also available as a standalone JAR file. You can use the Business Central standalone JAR file to run Business Central without needing to deploy it to an application server.

Red Hat JBoss Enterprise Application Platform (Red Hat JBoss EAP) 7.4 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 7.4 server instance.



NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

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

- *Installing and configuring KIE Server on IBM WebSphere Application Server*
- *Installing and configuring KIE Server on Oracle WebLogic Server*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 4 using Operators*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 3 using templates*

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

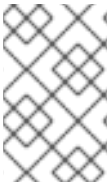
CHAPTER 11. RED HAT DECISION MANAGER ROLES AND USERS

To access Business Central or KIE Server, you must create users and assign them appropriate roles before the servers are started. You can create users and roles when you install Business Central or KIE Server.

If both Business Central and KIE Server are running on a single instance, a user who is authenticated for Business Central can also access KIE Server.

However, if Business Central and KIE Server are running on different instances, a user who is authenticated for Business Central must be authenticated separately to access KIE Server. For example, if a user who is authenticated on Business Central but not authenticated on KIE Server tries to view or manage process definitions in Business Central, a 401 error is logged in the log file and the **Invalid credentials to load data from remote server. Contact your system administrator.** message appears in Business Central.

This section describes Red Hat Decision Manager user roles.



NOTE

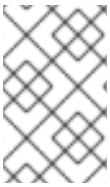
The **admin**, **analyst**, and **rest-all** roles are reserved for Business Central. The **kie-server** role is reserved for KIE Server. For this reason, the available roles can differ depending on whether Business Central, KIE Server, or both are installed.

- **admin**: Users with the **admin** role are the Business Central administrators. They can manage users and create, clone, and manage 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 Decision Manager.
- **analyst**: Users with the **analyst** role have access to all high-level features. They can model 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.
- **rest-all**: Users with the **rest-all** role can access Business Central REST capabilities.
- **kie-server**: Users with the **kie-server** role can access KIE Server REST capabilities.

CHAPTER 12. DOWNLOADING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION FILES

You can use the installer JAR file or deployable ZIP files to install Red Hat Process Automation Manager. You can run the installer in interactive or command line interface (CLI) mode. Alternatively, you can extract and configure the Business Central and KIE Server deployable ZIP files. If you want to run Business Central without deploying it to an application server, download the Business Central Standalone JAR file.

Download a Red Hat Process Automation Manager distribution that meets your environment and installation requirements.



NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

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:** 7.13.5
2. Download one of the following product distributions, depending on your preferred installation method:

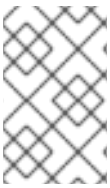
NOTE

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 7.4, download **Red Hat Process Automation Manager 7.13.5 Installer**(**rhpam-installer-7.13.5.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 7.4 using the deployable ZIP files, download the following files:
     - **Red Hat Process Automation Manager 7.13.5 KIE Server for All Supported EE8 Containers** (**rhpam-7.13.5-kie-server-ee8.zip**)
     - **Red Hat Process Automation Manager 7.13.5 KIE Server Deployable for EAP 7** (**rhpam-7.13.5-business-central-eap7-deployable.zip**)
   - To run Business Central without needing to deploy it to an application server, download **Red Hat Process Automation Manager 7.13.5 Business Central Standalone** (**rhpam-7.13.5-business-central-standalone.jar**).

CHAPTER 13. USING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLER

This section describes how to install Business Central and KIE Server 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 7.4 server installation. You can run the installer in interactive or command line interface (CLI) mode.



NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

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 7.13 Deployable for Red Hat JBoss EAP 7.4** file and follow the steps in Chapter 14, *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 13.1, “Using the installer in interactive mode”
- Section 13.2, “Using the installer in CLI mode”

13.1. 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 7.4 server installation.

NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

NOTE

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

Prerequisites

- The **Red Hat Process Automation Manager 7.13.5 Installer**has been downloaded. For instructions, see Chapter 12, *Downloading the Red Hat Process Automation Manager installation files*.
- 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 7.4 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. For a list of system properties, see Appendix A, *Business Central system properties* .

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-7.13.5.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-7.13.5.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-7.13.5.jar
```

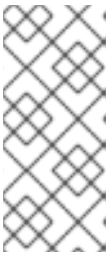
The graphical installer displays a splash screen and a license agreement page.

2. Click **I accept the terms of this license agreement**and click **Next**.
3. Specify the Red Hat JBoss EAP 7.4 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 KIE Server on the same server. However, you should install Business Central and KIE 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 KIE Server in the same container the new user is given the **admin**, **kie-server**, and **rest-all** roles. If you install only KIE Server, the user is given the **kie-server** role. To select another role, deselect **admin**. For information about roles, see Chapter 11, *Red Hat Decision 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 KIE Server.

6. On the **Installation Overview** page, review the components that you will install and click **Next** to start the installation.

7. When the installation has completed, click **Next**.

8. When **Processing finished** appears at the top of the screen, click **Next** to complete the installation.

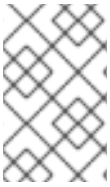
9. Optional: Click **Generate Installation Script and Properties File** to save the installation data in XML files, and then click **Done**.
The installer generates two files. The **auto.xml** file automates future installations and the **auto.xml.variables** file stores user passwords and other sensitive variables. Use the **auto.xml** file to repeat the Red Hat Process Automation Manager installation on multiple systems with the same type of server and the same configuration as the original installation. If necessary, update the **installpath** parameter in the **auto.xml** file. To perform an installation using the XML file, enter the following command:

```
java -jar rhpam-installer-7.13.5.jar <path-to-auto.xml-file>
```

You have successfully installed Red Hat Process Automation Manager using the installer.

13.2. USING THE INSTALLER IN CLI MODE

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



NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

NOTE

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

Prerequisites

- The **Red Hat Process Automation Manager 7.13.5 Installer**has been downloaded. For instructions, see Chapter 12, *Downloading the Red Hat Process Automation Manager installation files*.
- 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 7.4 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. For a list of system properties, see Appendix A, *Business Central system properties* .

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-7.13.5.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 or Red Hat JBoss Web Server 5. For more information, see https://access.redhat.com/articles/3405381[Red Hat Process Automation Manager 7 Supported Configurations].
```

3. Enter the parent directory of an existing Red Hat JBoss EAP 7.4 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 KIE Server on the same server. However, you should install Business Central and KIE 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 KIE 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.

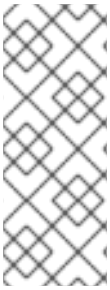
7. Enter a path or press the Enter key to accept the suggested path.
The installer generates two files. The **auto.xml** file automates future installations and the **auto.xml.variables** file stores user passwords and other sensitive variables. Use the **auto.xml** file on multiple systems to easily 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. To perform an installation using the XML file, enter the following command:

```
java -jar rhpam-installer-7.13.5.jar <path-to-auto.xml-file>
```

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

CHAPTER 14. INSTALLING RED HAT PROCESS AUTOMATION MANAGER FROM ZIP FILES

You can use the Red Hat Process Automation Manager ZIP files (one for Business Central and one for KIE Server) to install Red Hat Process Automation Manager without using the installer.



NOTE

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

For information about installing the headless Process Automation Manager controller, see Chapter 19, *Installing and running the headless Process Automation Manager controller*.

14.1. INSTALLING BUSINESS CENTRAL FROM THE ZIP FILE

Business Central is the graphical user interface where you create and manage business rules that KIE Server executes. You can use a deployable ZIP file to install and configure Business Central.

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.4 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 12, *Downloading the Red Hat Process Automation Manager installation files*:
  rhpam-7.13.5-business-central-eap7-deployable.zip

Procedure

1. Extract the **rhpam-7.13.5-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*/rhpam-7.13.5-business-central-eap7-deployable/jboss-eap-7.4** directory to ***EAP_HOME***. When prompted, merge or replace files.

WARNING

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

14.2. INSTALLING KIE SERVER FROM THE ZIP FILE

KIE Server provides the runtime environment for business assets and accesses the data stored in the assets repository (knowledge store). You can use a deployable ZIP file to install and configure KIE Server.

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.4 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 12, *Downloading the Red Hat Process Automation Manager installation files*:
  rhpam-7.13.5-kie-server-ee8.zip

Procedure

1. Extract the **rhpam-7.13.5-kie-server-ee8.zip** archive to a temporary directory. In the following examples this directory is called ***TEMP_DIR***.
2. Copy the ***TEMP_DIR*/rhpam-7.13.5-kie-server-ee8/kie-server.war** directory to ***EAP_HOME*/standalone/deployments/**.

WARNING

Ensure the names of the Red Hat Decision 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*/rhpam-7.13.5-kie-server-ee8/rhpam-7.13.5-kie-server-ee8/SecurityPolicy/** to ***EAP_HOME*/bin**.
4. When prompted to overwrite files, click **Replace**.
5. In the ***EAP_HOME*/standalone/deployments/** directory, create an empty file named **kie-server.war.dodeploy**. This file ensures that KIE Server is automatically deployed when the server starts.

14.3. CREATING USERS

If you used the deployable ZIP files to install Red Hat Process Automation Manager, before you can log in to Business Central or KIE 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 KIE Server user that has the **kie-server** role. For information about roles, see Chapter 11, *Red Hat Decision Manager roles and users* .

NOTE

Red Hat Decision Manager stores user data as a set of properties or as a set of files. File-based storage provides several extra features, such as SSH login and a user maintenance UI.

The user script examples in this documentation use the file-based user script, **jboss-cli.sh**, instead of the property-based user script, **add-users.sh**.

Prerequisites

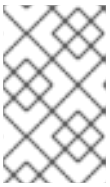
- Red Hat Process Automation Manager is installed in the base directory of the Red Hat JBoss EAP installation (***EAP_HOME***).

Procedure

1. Optional: To change Red Hat Process Automation Manager from using property-based user storage to file-based user storage, complete the following steps:
   a. Run the following command to apply the **kie-fs-realm** patch:

```
$ ./bin/elytron-tool.sh filesystem-realm --users-file application-users.properties --roles-file application-roles.properties --output-location kie-fs-realm-users
```

   b. Open each **kie-fs-realm-users**/*/**<USER>.xml** file where **<USER>** is a Red Hat Process Automation Manager user.
   c. Replace **<attribute name="roles" value=** with **<attribute name="role" value=**.
2. In a terminal application, navigate to the ***EAP_HOME*/bin** directory.
3. Create a user with the **admin**, **rest-all**, and **kie-server** roles.



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 KIE Server (KIE Server) REST capabilities.

In the following command, replace **<USERNAME>** and **<PASSWORD>** with the user name and password of your choice:

```
$ ./bin/jboss-cli.sh --commands="embed-server --std-out=echo,/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity(identity=<USERNAME>),/subsystem=elytron/filesystem-realm=ApplicationRealm:set-password(identity=<USERNAME>, clear={password='<PASSWORD>'}),/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity-attribute(identity=<USERNAME>, name=role, value=[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).

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

```
$ ./bin/jboss-cli.sh --commands="embed-server --std-
out=echo,/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity(identity=
<USERNAME>),/subsystem=elytron/filesystem-realm=ApplicationRealm:set-
password(identity=<USERNAME>, clear=
{password='<PASSWORD>'}),/subsystem=elytron/filesystem-realm=ApplicationRealm:add-
identity-attribute(identity=<USERNAME>, name=role, value=[kie-server])"
```

5. Make a note of your user names and passwords.

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

```
$ ./bin/jboss-cli.sh --commands="embed-server --std-
out=echo,/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity(identity=
<USERNAME>),/subsystem=elytron/filesystem-realm=ApplicationRealm:set-
password(identity=<USERNAME>, clear=
{password='<PASSWORD>'}),/subsystem=elytron/filesystem-realm=ApplicationRealm:add-
identity-attribute(identity=<USERNAME>, name=role, value=[admin,rest-all,kie-server])"
```

NOTE

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

7. Optional: To create several users at one time, create a file that contains the user data and run the following command, where **<USER_DATA>.cli** is the file that contains the user data:

```
$ ./bin/jboss-cli.sh --file=<USER_DATA>.cli
```

The **<USER_DATA>.cli** file should contain data similar to the following example:

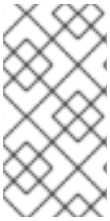
```
embed-server --std-out=echo

# first user
/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity(identity=<USERNAME>)
/subsystem=elytron/filesystem-realm=ApplicationRealm:set-password(identity=
<USERNAME>, clear={password='<PASSWORD>'})
/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity-attribute(identity=
<USERNAME>, name=role, value=[admin,role,group])

# second user
...
```

14.4. CONFIGURING KIE SERVER TO CONNECT TO BUSINESS CENTRAL

WARNING

This section provides a sample setup that you can use for testing purposes. Some of the values are unsuitable for a production environment, and are marked as such.

If a KIE Server is not configured in your Red Hat Process Automation Manager environment, or if you require additional KIE Servers in your Red Hat Process Automation Manager environment, you must configure a KIE Server to connect to Business Central.



NOTE

If you are deploying KIE Server on Red Hat OpenShift Container Platform, see the *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 4 using Operators* document for instructions about configuring it to connect to Business Central.

Prerequisites

- Business Central and KIE Server are installed in the base directory of the Red Hat JBoss EAP installation (***EAP_HOME***) as described in the following sections:
  - Section 14.1, "Installing Business Central from the ZIP file"
  - Section 14.2, "Installing KIE Server from the ZIP file"
- Users with the following roles exist:
  - In Business Central, a user with the role **rest-all**
  - On KIE Server, a user with the role **kie-server**
    For more information, see Section 14.3, "Creating users".

Procedure

1. 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**.
2. Open the **standalone-full.xml** file and under the **<system-properties>** tag, set the following JVM properties:

   Table 14.1. JVM Properties for the managed KIE Server instance

Property	Value	Note
org.kie.server.id	**default-kie-server**	The KIE Server ID.
org.kie.server.controller	**http://localhost:8080/business-central/rest/controller**	The location of Business Central. The URL for connecting to the API of Business Central.
org.kie.server.controller.user	**controllerUser**	The user name with the role **rest-all** who can log in to the Business Central.
org.kie.server.controller.pwd	**controllerUser1234;**	The password of the user who can log in to the Business Central.
org.kie.server.location	**http://localhost:8080/kie-server/services/rest/server**	The location of KIE Server. The URL for connecting to the API of KIE Server.

Table 14.2. JVM Properties for the Business Central instance

Property	Value	Note
org.kie.server.user	**controllerUser**	The user name with the role **kie-server**.
org.kie.server.pwd	**controllerUser1234;**	The password of the user.

The following example shows how to configure a KIE Server instance:

```
<property name="org.kie.server.id" value="default-kie-server"/>
<property name="org.kie.server.controller" value="http://localhost:8080/business-central/rest/controller"/>
<property name="org.kie.server.controller.user" value="controllerUser"/>
<property name="org.kie.server.controller.pwd" value="controllerUser1234;"/>
<property name="org.kie.server.location" value="http://localhost:8080/kie-server/services/rest/server"/>
```

The following example shows how to configure a for Business Central instance:

```
<property name="org.kie.server.user" value="controllerUser"/>
<property name="org.kie.server.pwd" value="controllerUser1234;"/>
```

3. To verify that KIE Server starts successfully, send a GET request to **http://*SERVER:PORT*/kie-server/services/rest/server/** when KIE Server is running. For more information about running Red Hat Process Automation Manager on KIE Server, see *Running Red Hat Process Automation Manager*.
   After successful authentication, you receive an XML response similar to the following example:

```
<response type="SUCCESS" msg="Kie Server info">
  <kie-server-info>
    <capabilities>KieServer</capabilities>
    <capabilities>BRM</capabilities>
    <capabilities>BPM</capabilities>
    <capabilities>CaseMgmt</capabilities>
    <capabilities>BPM-UI</capabilities>
    <capabilities>BRP</capabilities>
    <capabilities>DMN</capabilities>
    <capabilities>Swagger</capabilities>
    <location>http://localhost:8230/kie-server/services/rest/server</location>
    <messages>
      <content>Server KieServerInfo{serverId='first-kie-server', version='7.5.1.Final-redhat-
1', location='http://localhost:8230/kie-server/services/rest/server', capabilities=[KieServer,
BRM, BPM, CaseMgmt, BPM-UI, BRP, DMN, Swagger]}started successfully at Mon Feb 05
15:44:35 AEST 2018</content>
      <severity>INFO</severity>
      <timestamp>2018-02-05T15:44:35.355+10:00</timestamp>
    </messages>
    <name>first-kie-server</name>
    <id>first-kie-server</id>
    <version>7.5.1.Final-redhat-1</version>
  </kie-server-info>
</response>
```

4. Verify successful registration:

   a. Log in to Business Central.

   b. Click **Menu → Deploy → Execution Servers**.
   If registration is successful, you will see the registered server ID.

14.5. THREAD EFFICIENCY

To ensure that the optimal number of threads are used, set the value of the threading system properties to the sum of the number of CPUs plus one.

1. 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**.

2. Open the **standalone-full.xml** file.

3. Under the **<system-properties>** tag, set the value of the following JVM properties to the number of CPUs plus one:

```
org.appformer.concurrent.managed.thread.limit
org.appformer.concurrent.unmanaged.thread.limit
org.appformer.concurrent.indexing.thread.limit
org.appformer.concurrent.rest.api.thread.limit
```

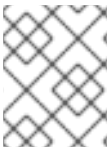
NOTE

The number of CPUs plus one is a valid baseline value for all properties. You might have to fine-tune further based on additional testing.

CHAPTER 15. SECURING PASSWORDS WITH A KEYSTORE

You can use a keystore to encrypt passwords that are used for communication between Business Central and KIE Server. You should encrypt both controller and KIE Server passwords. If Business Central and KIE 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 KIE Server is not configured with JCEKS, KIE Server passwords are stored in system properties in plain text form.

Prerequisites

- KIE 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 KIE Server user with the **kie-server** role and specify a password. In the following example, replace **<USERNAME>** and **<PASSWORD>** with the user name and password of your choice.

```
$<EAP_HOME>./bin/jboss-cli.sh --commands="embed-server --std-
out=echo,/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity(identity=
<USERNAME>),/subsystem=elytron/filesystem-realm=ApplicationRealm:set-
password(identity=<USERNAME>, clear=
{password='<PASSWORD>'}),/subsystem=elytron/filesystem-realm=ApplicationRealm:add-
identity-attribute(identity=<USERNAME>, name=role, value=['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 KIE 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>"/>
  <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 15.1. System properties used to load a KIE 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_ALIAS>**	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_ALIAS>**	Alias of the key for default REST Process Automation Controller where the password is stored
kie.keystore.key.ctrl.pwd	**<KEY_CONTROL_PWD>**	Password of the alias for default REST Process Automation Controller with the stored password

5. Start KIE Server to verify the configuration.

CHAPTER 16. 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 Decision Manager 7 Supported Configurations.

Procedure

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

- Modify the **~/.ssh/config** file on the 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
```

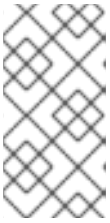
CHAPTER 17. VERIFYING THE RED HAT DECISION MANAGER INSTALLATION

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 KIE 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 you configured Red Hat Process Automation Manager to run from a domain name, replace **localhost** with the domain name, for example: **http://www.example.com:8080/business-central**
   - If you configured Red Hat Process Automation Manager 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**.
7. Open the **MySpace** space.
8. Click **Try Samples → Mortgages → OK**. The **Assets** window appears.
9. Click **Add Asset → Data Object**.
10. Enter **MyDataObject** in the **Data Object** field and click **OK**.
11. Click **Spaces → MySpace → Mortgages** and confirm that **MyDataObject** is in the list of assets.
12. Delete the **Mortgages** project.

CHAPTER 18. RUNNING RED HAT PROCESS AUTOMATION MANAGER

Use this procedure to run 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.

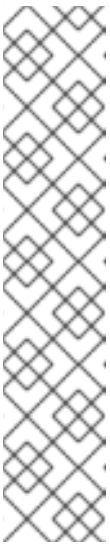
Procedure

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 KIE 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**.
   If you configured Red Hat Process Automation Manager to run from a domain name, replace **localhost** with the domain name, for example:

   http://www.example.com:8080/business-central

4. Log in using the credentials of the user that you created for Business Central in Section 14.3, “Creating users”.

CHAPTER 19. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

You can configure KIE Server to run in managed or unmanaged mode. If KIE Server is unmanaged, you must manually create and maintain KIE containers (deployment units). If KIE Server is managed, the Process Automation Manager controller manages the KIE 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 KIE Server management without Business Central, you can use the headless Process Automation Manager controller.

19.1. INSTALLING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

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

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.4 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:** 7.13.5
2. Download **Red Hat Process Automation Manager 7.13.5 Add Ons**(the **rhpam-7.13.5-add-ons.zip** file).
3. Extract the **rhpam-7.13.5-add-ons.zip** file. The **rhpam-7.13.5-controller-ee7.zip** file is in the extracted directory.
4. Extract the **rhpam-7.13.5-controller-ee7.zip** archive to a temporary directory. In the following examples this directory is called ***TEMP_DIR***.
5. Copy the ***TEMP_DIR*/rhpam-7.13.5-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-7.13.5-controller-ee7/SecurityPolicy/** directory to ***EAP_HOME*/bin**.
7. When prompted to overwrite files, select **Yes**.
8. 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.

19.1.1. 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.

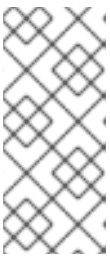
Prerequisites

- 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 **<USERNAME>** and **<PASSWORD>** with the user name and password of your choice.

```
$ ./bin/jboss-cli.sh --commands="embed-server --std-
out=echo,/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity(identity=
<USERNAME>),/subsystem=elytron/filesystem-realm=ApplicationRealm:set-
password(identity=<USERNAME>, clear=
{password='<PASSWORD>'}),/subsystem=elytron/filesystem-realm=ApplicationRealm:add-
identity-attribute(identity=<USERNAME>, name=role, value=['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).

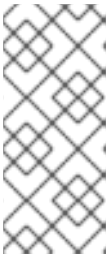
3. Make a note of your user name and password.

19.1.2. Configuring KIE Server and the headless Process Automation Manager controller

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

Prerequisites

- KIE Server is installed in an ***EAP_HOME***.
- The headless Process Automation Manager controller is installed in an ***EAP_HOME***.



NOTE

You should install KIE Server and the headless Process Automation Manager controller on different servers in production environments. However, if you install KIE 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 KIE 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 14.3, “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 KIE 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 KIE Server installation, for example, **rhpam-7.13.5-kie-server-1**.

- Replace **<HOST>** with the ID or name of the KIE Server host, for example, **localhost** or **192.7.8.9**.
- Replace **<PORT>** with the port of the KIE Server host, for example, **8080**.

NOTE

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

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

19.2. 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 KIE 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 KIE Server, 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:

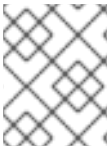
     ```
     $ ./standalone.sh
     ```

   - 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 19.1.1, “Creating a headless Process Automation Manager controller user” . The output of this command provides information about the KIE Server instance.

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

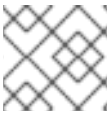


NOTE

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

CHAPTER 20. CONFIGURING AND RUNNING STANDALONE BUSINESS CENTRAL

You can use the Business Central standalone JAR file to run Business Central without deploying it to an application server. You can use sample configuration files to run the Business Central standalone JAR file out of the box or you can customize the sampfiles for your requirements.



NOTE

This JAR file is supported only when it is run on Red Hat Enterprise Linux.

Prerequisites

- The **Red Hat Process Automation Manager 7.13.5 Business Central Standalone**(**rhpam-7.13.5-business-central-standalone.jar**) and the **Red Hat Process Automation Manager 7.13.5 Add Ons** (**rhpam-7.13.5-add-ons.zip**) files have been downloaded from the Software Downloads page for Red Hat Process Automation Manager 7.13, as described in Chapter 12, *Downloading the Red Hat Process Automation Manager installation files* .

Procedure

1. Extract the downloaded **rhpam-7.13.5-addons.zip** to a temporary directory. This archive includes the **rhpam-7.13.5-standalone-sample-configuration.zip** file.
2. Extract the **rhpam-7.13.5-standalone-sample-configuration.zip** file to the directory that contains the **rhpam-7.13.5-business-central-standalone.jar** file. The **rhpam-7.13.5-standalone-sample-configuration.zip** file contains the following sample configuration files:
   - **application-script.cli**: Sample script for adding a user and kie server system properties
   - **kie-fs-realm-users**: Sample user data
     You can run the **rhpam-7.13.5-business-central-standalone.jar** files with the sample data provided in the configuration files or you can customize the data for your requirements.
3. To customize the configuration data, complete the following steps:
   a. Edit the **application-script.cli** file to include an administrative user with **admin**, **user**, **rest-all**, **rest-client** and **kie-server** roles. In the following example, replace **<USERNAME>** and **<PASSWORD>** with your username and password of the user you want to create.

```
/subsystem=elytron/filesystem-realm=KieRealm:add-identity(identity=<USERNAME>)
/subsystem=elytron/filesystem-realm=KieRealm:set-password(identity=<USERNAME>,
clear={password="<PASSWORD>"})
/subsystem=elytron/filesystem-realm=KieRealm:add-identity-attribute(identity=
<USERNAME>, name=role, value=["admin","user","rest-all","rest-client","kie-server"])
```

4. To run the Business Central standalone JAR file, enter the following command:

```
java -jar rhpam-7.13.5-business-central-standalone.jar --cli-script=application-script.cli
```

5. To set application properties when you run the JAR file, include the **-D<PROPERTY>=<VALUE>** parameter in the command, where **<PROPERTY>** is the name of a supported application property and **<VALUE>** is the property value:

```
java -jar rhpam-7.13.5-business-central-standalone.jar --cli-script=application-script.cli -
D<PROPERTY>=<VALUE> -D<PROPERTY>=<VALUE>
```

For example, to run Business Central and connect to KIE Server as the user **controllerUser**, enter:

```
java -jar rhpam-7.13.5-business-central-standalone.jar \
 --cli-script=application-script.cli \
-Dorg.kie.server.user=controllerUser \
-Dorg.kie.server.pwd=controllerUser1234
```

Doing this enables you to deploy containers to KIE Server. See Appendix A, *Business Central system properties* for more information.

NOTE

To enable user and group management in Business Central, set the value of the **org.uberfire.ext.security.management.wildfly.cli.folderPath** property to **kie-fs-realm-users**.

CHAPTER 21. MAVEN SETTINGS AND REPOSITORIES FOR RED HAT DECISION MANAGER

When you create a Red Hat Decision 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 Decision Manager projects to retrieve dependencies from the public Red Hat Decision Manager repository by modifying the Maven project object model (POM) file (**pom.xml**). You can also configure Business Central and KIE Server to use an external Maven repository or prepare a Maven mirror for offline use.

For more information about Red Hat Decision Manager packaging and deployment options, see *Packaging and deploying an Red Hat Decision Manager project* .

21.1. ADDING MAVEN DEPENDENCIES FOR RED HAT DECISION MANAGER

To use the correct Maven dependencies in your Red Hat Decision 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>7.13.5.redhat-00002</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 dependencies are provided automatically by Business Central.

    - For a basic Red Hat Decision Manager project, declare the following dependencies

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

- For a basic Red Hat Decision 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 KIE Server, declare the following dependencies:

Client application KIE Server dependencies

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

- To create a remote client for Red Hat Decision 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 Decision 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>
```

21.2. CONFIGURING AN EXTERNAL MAVEN REPOSITORY FOR BUSINESS CENTRAL AND KIE SERVER

You can configure Business Central and KIE Server to use an external Maven repository, such as Nexus or Artifactory, instead of the built-in repository. This enables Business Central and KIE 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 because 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. The version numbers of patched artifacts are 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 an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 4 using Operators*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 3 using templates*

Prerequisites

- Business Central and KIE Server are installed. For installation options, see *Planning a Red Hat Decision 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"/>
```

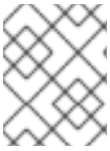
5. Start or restart Business Central and KIE Server.

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 an Red Hat Decision Manager project* .

21.3. 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:
2. 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:** 7.13.5
     a. Download and extract the **Red Hat Process Automation Manager 7.13.5 Offliner Content List** (**rhpam-7.13.5-offliner.zip**) product deliverable file.
     b. Extract the contents of the **rhpam-7.13.5-offliner.zip** file into any directory.

c. Change to the directory and enter the following command:

```
./offline-repo-builder.sh offliner.txt
```

This command creates the **repository** subdirectory and downloads the necessary artifacts into this subdirectory. This is the mirror repository.

If a message reports that some downloads have failed, run the same command again. If downloads fail again, contact Red Hat support.

3. If you developed services outside of 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 path of the **pom.xml** file of the project.

d. Copy the contents of the local Maven cache directory (**~/.m2/repository**) to the **repository** subdirectory that was created.

4. Copy the contents of the **repository** subdirectory to a directory on the computer on which you deployed Red Hat Process Automation Manager. This directory becomes the offline Maven mirror repository.

5. Create and configure a **settings.xml** file for your Red Hat Process Automation Manager deployment as described in Section 21.2, "Configuring an external Maven repository for Business Central and KIE Server".

6. Make the following changes in the **settings.xml** file:

- Under the **<profile>** tag, if a **<repositories>** or **<pluginRepositores>** tag is missing, add the missing tags.

- Under **<repositories>** add the following content:

```
<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 content:

```
<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.

7. Set the **kie.maven.offline.force** property for Business Central to **true**. For instructions about setting properties for Business Central, see *Installing and configuring Red Hat Decision Manager on Red Hat JBoss EAP 7.4*.

CHAPTER 22. 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 Decision 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. To import a project, do one of the following:
   - Click **Import Project**.
   - Select **Import Project** from the drop-down list.
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 23. 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.

23.1. 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 an SVG format image to the ***EAP_HOME*/standalone/deployments/business-central.war/img/** directory in your Red Hat Decision Manager installation.
3. In the ***EAP_HOME*/standalone/deployments/business-central.war/img/** directory, either move or rename the existing **redhat_logo.png** file.
4. Rename your PNG file **redhat_logo.png**.
5. To change the project logo that appears above the **User name** and **Password** fields, replace the default image **BC_Logo.png** 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.

23.2. 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 Decision Manager installation.
3. Open the ***EAP_HOME*/standalone/deployments/business-central.war/banner/banner.html** file in a text editor.
4. Replace **logo.png** in the **<img>** tag with the file name of your new image:admin-and-config/

```
<img src="banner/logo.png"/>
```

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 24. INTEGRATING LDAP AND SSL

With Red Hat Process Automation Manager you can integrate LDAP and SSL through Red Hat Single Sign-On. For more information, see the *Red Hat Single Sign-On Server Administration Guide* .

APPENDIX A. BUSINESS CENTRAL SYSTEM PROPERTIES

The Business Central system properties listed in this section are passed to **standalone*.xml** files.

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 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 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.

KIE Server nodes and Process Automation Manager controller

Use the following properties to configure the connections with the KIE 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 KIE 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 KIE 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.builder.cache.size**: Defines the cache size of the project builder. Default value: **20**.
- **org.kie.library.assets_per_page**: You can customize the number of assets per page in the project screen. Default value: **15**.
- **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 KIE Server rendered forms:

- **org.jbpm.wb.forms.renderer.ext**: Switches the form rendering between Business Central and KIE 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 KIE Server rendered forms. Default value: **workbench**.

CHAPTER 25. ADDITIONAL RESOURCES

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

CHAPTER 26. NEXT STEPS

- *Integrating Red Hat Decision Manager with Red Hat Single Sign-On*
- *Getting started with decision services*

PART III. INSTALLING AND CONFIGURING RED HAT PROCESS AUTOMATION MANAGER IN A RED HAT JBOSS EAP CLUSTERED ENVIRONMENT

As a system engineer, you can create a Red Hat Process Automation Manager clustered environment to provide high availability and load balancing for your development and runtime environments.

Prerequisites

- You have reviewed the information in *Planning a Red Hat Decision Manager installation* .

CHAPTER 27. RED HAT DECISION MANAGER CLUSTERS

By clustering two or more computers, you have the benefits of high availability, enhanced collaboration, and load balancing. High availability decreases the chance of data loss when a single computer fails. When a computer fails, another computer fills the gap by providing a copy of the data that was on the failed computer. When the failed computer comes online again, it resumes its place in the cluster.

There are several ways that you can cluster Red Hat Decision Manager components. This document describes how to cluster the following scenarios:

- Chapter 28, *Red Hat Process Automation Manager clusters in a development (authoring) environment*
- Chapter 29, *KIE Server clusters in a runtime environment*

CHAPTER 28. RED HAT PROCESS AUTOMATION MANAGER CLUSTERS IN A DEVELOPMENT (AUTHORING) ENVIRONMENT

You can configure Red Hat Process Automation Manager as a clustered development environment to benefit from high availability. With a clustered environment, if a developer is working on a node and that node fails, that developer's work is preserved and visible on any other node of the cluster.

Most development environments consist of Business Central and at least one KIE Server .

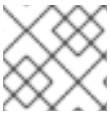
To create a Red Hat Decision Manager clustered development environment, you must perform the following tasks:

1. Configure the following components on each system that will be a node of the cluster:
   - Configure Red Hat JBoss EAP 7.4 with Red Hat Data Grid 8.1.
     Red Hat Data Grid is built from the Infinispan open-source software project. It is a distributed in-memory key/value data store that has indexing capabilities that enable you to store, search, and analyze high volumes of data quickly and in near-real time. In a Red Hat Decision Manager clustered environment, it enables you to perform complex and efficient searches across cluster nodes.
   - Configure AMQ Broker, a Java messaging server (JMS) broker.
     A JMS broker is a software component that receives messages, stores them locally, and forwards the messages to a recipient. AMQ Broker enables your applications to communicate with any messaging provider. It specifies how components such as message-driven beans, Enterprise JavaBeans, and servlets can send or receive messages.
   - Configure an NFS file server.
2. Download Red Hat JBoss EAP 7.4 and Red Hat Decision Manager 7.13, and then install them on each system that will be a cluster node.
3. Configure and start Business Central on each node of the cluster.

28.1. INSTALLING AND CONFIGURING RED HAT DATA GRID

For more efficient searching across cluster nodes, install Red Hat Data Grid and configure it for the Red Hat Decision Manager clustered environment.

For information about Red Hat Data Grid advanced installation and configuration options and Red Hat Data Grid modules for Red Hat JBoss EAP, see the *Red Hat Data Grid Server Guide* .



NOTE

Do not install Red Hat Data Grid on the same node as Business Central.

Prerequisites

- A Java Virtual Machine (JVM) environment compatible with Java 8.0 or later is installed.

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:** Data Grid
   - **Version:** 8.1
2. Download and extract the **Red Hat Data Grid 8.1.0 Server**(**redhat-datagrid-8.1.0-server.zip**) installation file to the preferred location on your system. In the following examples, the extracted directory is referred to as ***JDG_HOME***.
3. Update Red Hat Data Grid to the latest version. For more information, see Red Hat Data Grid *Red Hat Data Grid User Guide* .
4. To add a Red Hat Data Grid user, navigate to ***JDG_HOME*/bin** and enter the following command:

```
$ ./cli.sh user create <DATAGRID_USER_NAME> -p <DATA_GRID_PASSWORD> -r
default
```

5. To change Red Hat Data Grid server memory parameters, open the ***JDG_HOME*/bin/server.conf** file and locate the following line:

```
-Xms64m -Xmx512m -XX:MetaspaceSize=64M
```

6. Replace this line with the following content:

```
-Xms256m -Xmx2048m -XX:MetaspaceSize=256M
```

7. Open ***JDG_HOME*/server/conf/infinispan.xml** file and locate the following line:

```
<hotrod-connector name="hotrod"/>
```

8. Replace this line with the following content:

```
<hotrod-connector name="hotrod">
   <authentication>
      <sasl mechanisms="SCRAM-SHA-512 SCRAM-SHA-384 SCRAM-SHA-256
            SCRAM-SHA-1 DIGEST-SHA-512 DIGEST-SHA-384
            DIGEST-SHA-256 DIGEST-SHA DIGEST-MD5 PLAIN"
         server-name="infinispan"
         qop="auth"/>
   </authentication>
</hotrod-connector>
```

9. To run Red Hat Data Grid, navigate to ***JDG_HOME*** and enter the following command:

```
$ ./server.sh -b <HOST>
```

   Replace **<HOST>** with the IP address or host name of the server where you installed Red Hat Data Grid.

28.2. DOWNLOADING AND CONFIGURING AMQ BROKER

Red Hat AMQ Broker enables your applications to communicate with any messaging provider. It specifies how components such as message-driven beans, Enterprise JavaBeans, and servlets can send or receive messages.

For information about advanced installations and configuration options, see *Getting started with AMQ Broker*.

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: AMQ Broker**
   - **Version: 7.8.2**
2. Click **Download** next to **Red Hat AMQ Broker 7.8.2**(**amq-broker-7.8.2-bin.zip**).
3. Extract the **amq-broker-7.8.2-bin.zip** file.
4. Change directory to **amq-broker-7.8.2-bin/amq-broker-7.8.2/bin**.
5. Enter the following command to create the broker and broker user:

```
./artemis create --host <HOST> --user <AMQ_USER> --password <AMQ_PASSWORD> --require-login <BROKER_NAME>
```

   In this example, replace the following placeholders:
   - **<HOST>** is the IP address or host name of the server where you installed AMQ Broker.
   - **<AMQ_USER>** and **<AMQ_PASSWORD>** is a user name and password combination of your choice.
   - **<BROKER_NAME>** is a name for the broker that you are creating.
6. To run AMQ Broker, navigate to the **amq-broker-7.8.2-bin/amq-broker-7.8.2/bin** directory and enter the following command:

```
<BROKER_NAME>/bin/artemis run
```

28.3. CONFIGURING AN NFS VERSION 4 SERVER

An NFS version 4 server with a shared file system is required for a Business Central clustered environment and each client node must have access to the shared file system.

Procedure

1. Configure a server to export NFS version 4 shares. For instructions about exporting NFS shares on Red Hat Enterprise Linux, see *Exporting NFS shares* in *Managing file systems*. For more information about creating the NFS server, see *How to configure NFS in RHEL 7*.
2. On the server, open the **/etc/exports** file in a text editor.

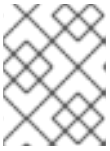
3. Add the following line to the **/etc/exports** file where **<HOST_LIST>** is a space-separated list of IP addresses and options of hosts that are authorized to connect to the server:

```
/opt/kie/data <HOST_LIST>
```

For example:

```
/opt/kie/data 192.168.1.0/24(rw,sync) 192.168.1.1/24(no_root_squash)
```

This creates an **/opt/kie/data** share with the **rw,sync,no_root_squash** minimum options that are required for NFS.



NOTE

You can use a different share name instead of **/opt/kie/data**. If you do, you must use the different name when configuring all nodes that run Business Central.

28.4. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.4 AND RED HAT PROCESS AUTOMATION MANAGER

Download and install Red Hat JBoss EAP 7.4 and Red Hat Process Automation Manager 7.13 on each node of the cluster.

Procedure

1. Download Red Hat JBoss EAP 7.4 on each node of the cluster:
   a. 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: Enterprise Application Platform**
      - **Version: 7.4**
   b. Click **Download** next to **Red Hat JBoss Enterprise Application Platform 7.4.14** (**JBEAP-7.4.14/jboss-eap-7.4.14.zip**).
2. Extract the **jboss-eap-7.4.14.zip** file. In the following steps, ***EAP_HOME*** is the **jboss-eap-7.4/jboss-eap-7.4** directory.
3. Download and apply the latest Red Hat JBoss EAP patch, if available.
4. Download Red Hat Process Automation Manager on each node of the cluster:
   a. Navigate to the Software Downloads page in the Red Hat Customer Portal, and select the product and version from the drop-down options:
      - **PRODUCT:** Process Automation Manager
      - **Version:** 7.13.5
   b. Download **Red Hat Decision Manager 7.13.5** Business Central Deployable for Red Hat JBoss EAP 7 (**rhpam-7.13.5-business-central-eap7-deployable.zip**).

5. Extract the **rhpam-7.13.5-business-central-eap7-deployable.zip** file to a temporary directory. In the following commands this directory is called ***TEMP_DIR***.

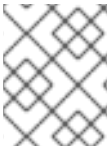
6. Copy the contents of ***TEMP_DIR*/rhpam-7.13.5-business-central-eap7-deployable/jboss-eap-7.4** to ***EAP_HOME***.

7. Download and apply the latest Red Hat Process Automation Manager patch, if available.

8. Configure Red Hat Single Sign-On for your high availability environment. For more information, see *Integrating Red Hat Decision Manager with Red Hat Single Sign-On* and the *Red Hat Single Sign-On Server Administration Guide*.

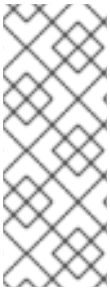
28.5. CONFIGURING AND RUNNING BUSINESS CENTRAL IN A CLUSTER

After you install Red Hat JBoss EAP and Business Central you can use Red Hat Data Grid and the AMQ Broker to configure the cluster. Complete these steps on each node of the cluster.



NOTE

These steps describe a basic cluster configuration. For more complex configurations, see the *Red Hat JBoss EAP 7.4 Configuration Guide* .



NOTE

Do not connect KIE Server to Business Central in high availability (HA) on premise environments.

Business Central instances are not able to keep in sync with the status of each KIE Server. For example, if a KIE Server is up but not in sync, Business Central will not be able to deploy through that instance.

Prerequisites

- Red Hat Data Grid 8.1 is installed as described in Section 28.1, "Installing and configuring Red Hat Data Grid".
- AMQ Broker is installed and configured, as described in Section 28.2, "Downloading and configuring AMQ Broker".
- Red Hat JBoss EAP and Red Hat Process Automation Manager are installed on each node of the cluster as described in Section 28.4, "Downloading and extracting Red Hat JBoss EAP 7.4 and Red Hat Process Automation Manager".
- An NFS server with a shared folder is available as described in Section 28.3, "Configuring an NFS version 4 server".

Procedure

1. To mount the directory shared over NFS as **/data**, enter the following commands as the root user:

```
mkdir /data
mount <NFS_SERVER_IP>:<DATA_SHARE> /data
```

Replace **<NFS_SERVER_IP>** with the IP address or hostname of the NFS server system. Replace **<DATA_SHARE>** with the share name that you configured (for example, **/opt/kie/data**).

2. Create a **kie-wb-playground** directory in the **/data** NFS directory:

```
mkdir /kie-wb-playground
```

3. Create a **kie-wb-playground** directory in the ***EAP_HOME*/bin** directory and mount the directory:

```
mount -o rw,sync,actimeo=1 <NFS_SERVER_IP>:<DATA_SHARE>/kie-wb-playground kie-wb-playground
```

4. Open the ***EAP_HOME*/standalone/configuration/standalone-full.xml** file in a text editor.

5. Edit or add the properties under the **<system-properties>** element and replace the following placeholders:

- **<AMQ_USER>** and **<AMQ_PASSWORD>** are the credentials that you defined when creating the AMQ Broker.
- **<AMQ_BROKER_IP_ADDRESS>** is the IP address of the AMQ Broker.
- **<DATA_GRID_NODE_IP>** is the IP address where Red Hat Data Grid is installed.
- **<SERVER_NAME>** is the server name specified in your Red Hat Data Grid server configuration.
- **<SASL_QOP>** is the combination of **auth**, **auth-int** and **auth-conf** values for your Red Hat Data Grid server configuration.
- **<DATAGRID_USER_NAME>** and **<DATA_GRID_PASSWORD>** are the credentials that you defined when creating the Red Hat Data Grid.

```
<system-properties>
  <property name="appformer-jms-connection-mode" value="REMOTE"/>
  <property name="appformer-jms-username" value="<AMQ_USER>"/>
  <property name="appformer-jms-password" value="<AMQ_USER_PASSWORD>"/>
  <property name="appformer-jms-url"
    value="tcp://<AMQ_BROKER_IP_ADDRESS>:61616?
ha=true&amp;retryInterval=1000&amp;retryIntervalMultiplier=1.0&amp;reconnectAttempts=
-1"/>
  <property name="org.appformer.ext.metadata.infinispan.port"
    value="11222"/>
  <property name="org.appformer.ext.metadata.infinispan.host"
    value="<DATA_GRID_NODE_IP>"/>
  <property name="org.appformer.ext.metadata.infinispan.realm"
    value="ApplicationRealm"/>
  <property name="org.appformer.ext.metadata.infinispan.cluster"
    value="kie-cluster"/>
  <property name="org.appformer.ext.metadata.index"
    value="infinispan"/>
  <property name="org.uberfire.nio.git.dir"
    value="/data"/>
  <property name="es.set.netty.runtime.available.processors"
```

```
    value="false"/>
  <property name="org.appformer.ext.metadata.infinispan.username" value="
<DATAGRID_USER_NAME>"/>
  <property name="org.appformer.ext.metadata.infinispan.password" value="
<DATA_GRID_PASSWORD>"/>
  <property name="org.appformer.ext.metadata.index" value="infinispan"/>
  <property name="org.appformer.ext.metadata.infinispan.sasl.qop" value="auth"/>
  <property name="org.appformer.ext.metadata.infinispan.server.name"
value="infinispan"/>
  <property name="org.appformer.ext.metadata.infinispan.realm" value="default"/>
  <property name="org.appformer.concurrent.managed.thread.limit" value="1000"/>
  <property name="org.appformer.concurrent.unmanaged.thread.limit" value="1000"/>
  <property name="org.appformer.concurrent.indexing.thread.limit" value="0"/>
  <property name="org.appformer.ext.metadata.infinispan.server.name"
    value="<SERVER_NAME>"/>
  <property name="org.appformer.ext.metadata.infinispan.sasl.qop"
    value="<SASL_QOP>"/>
</system-properties>
```

6. Save the **standalone-full.xml** file.
7. To start the cluster, navigate to ***EAP_HOME*/bin** and enter the following command:

```
$ ./standalone.sh -c standalone-full.xml -b <HOST>
```

Replace **<HOST>** with the IP address or host name of the server where you installed Red Hat Process Automation Manager.

28.6. TESTING YOUR HIGH AVAILABILITY (HA) ON-PREMISE INFRASTRUCTURE

When you create a production-ready high availability (HA) on-premises infrastructure for Business Central, you must ensure that it meets the minimum hardware and performance requirements for a viable HA environment. An HA on-premise infrastructure consists of the following four main components: Business Central, the message system (AMQ), the indexing server (Red Hat Data Grid), and a shared file system (NFS/GlusterFS/Ceph).

Prerequisites

- A network environment of at least 3 nodes is configured with the following layout:
  Node 1: Business Central

  Node 2: Business Central

  Node 3: AMQ, Red Hat Data Grid, and NFS

Procedure

1. Test the network speed:
   a. In the command terminal of each server node, install **iPerf3**:

```
$ dnf install iperf3
```

b. In the command terminal of the NFS server node (server node 3), start **iPerf3** in server mode:

```
$ iperf3 -s
```

c. In the command terminal of each Business Central server node, enter the following command to start **iPerf3** in client mode with the NFS server node set as the host:

```
$ iperf3 -c <NFS_SERVER_IP>
+
In this example, replace `<NFS_SERVER_IP>` with the IP address of the NFS server.
```

d. Compare the results from each server node with the following example of minimum values:

```
iperf3 -c 172.31.47.103
Connecting to host 172.31.47.103, port 5201
[  5] local 172.31.39.4 port 44820 connected to 172.31.47.103 port 5201
[ ID] Interval           Transfer     Bitrate         Retr  Cwnd
[  5]   0.00-1.00   sec   143 MBytes  1.20 Gbits/sec    0    419 KBytes
[  5]   1.00-2.00   sec   111 MBytes   928 Mbits/sec    6    848 KBytes
[  5]   2.00-3.00   sec  53.8 MBytes   451 Mbits/sec    0   1.08 MBytes
[  5]   3.00-4.00   sec  52.5 MBytes   440 Mbits/sec    1   1022 KBytes
[  5]   4.00-5.00   sec  53.8 MBytes   451 Mbits/sec    1    935 KBytes
[  5]   5.00-6.00   sec  53.8 MBytes   451 Mbits/sec    1    848 KBytes
[  5]   6.00-7.00   sec  52.5 MBytes   440 Mbits/sec    0   1.08 MBytes
[  5]   7.00-8.00   sec  53.8 MBytes   451 Mbits/sec    1   1.01 MBytes
[  5]   8.00-9.00   sec  53.8 MBytes   451 Mbits/sec    1    953 KBytes
[  5]   9.00-10.00  sec  52.5 MBytes   440 Mbits/sec    1    856 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-10.00  sec   680 MBytes   570 Mbits/sec   12             sender
[  5]   0.00-10.04  sec   677 MBytes   566 Mbits/sec                  receiver

iperf Done.
```

2. Verify the NFS information:

a. In the command terminal of each Business Central server node, mount the NFS node:

```
$ mount -o actimeo=1 <NFS_SERVER_IP>:/opt/nfs/kie /opt/kie/niogit
```

b. In the command terminal of each mounted node, enter **nfsiostat**:

```
$ nfsiostat
```

c. Compare the results from each server node with the following example of minimum values:

```
nfsiostat
ops/s rpc bklog
6.415 0.000

read:
ops/s kB/s kB/op retrans avg RTT (ms) avg exe (ms) avg queue (ms) errors
0.031 0.045 1.452 0 (0.0%) 0.129  0.166  0.019  0 (0.0%)
```

```
write:
ops/s kB/s kB/op retrans avg RTT (ms) avg exe (ms) avg queue (ms) errors
0.517 0.467 0.903 0 (0.0%) 1.235  1.269  0.01 8  0 (0.0%)
```

3. Verify that the disk is an SSD:

   a. In the command terminal of the NFS server, enter **df -h** to identify the disk as shown in the following example:

```
$ df -h
Filesystem              Size  Used Avail Use% Mounted on
devtmpfs                3.8G     0  3.8G   0% /dev
tmpfs                   3.9G     0  3.9G   0% /dev/shm
tmpfs                   3.9G   33M  3.8G   1% /run
tmpfs                   3.9G     0  3.9G   0% /sys/fs/cgroup
/dev/xvda2               25G  3.2G   22G  13% /
tmpfs                   781M     0  781M   0% /run/user/1000
172.31.47.103:/root/nfs   25G  2.1G   23G   9% /root/nfs
```

   b. Enter **lsblk -d** to verify that the disk is an SSD:

```
$ lsblk -d
```

   c. Enter **hdparm -Tt** to test the disk:

```
$ hdparm -Tt /dev/xvda2
```

   d. Compare the results from each server node with the following example of minimum values:

```
$ hdparm -Tt /dev/xvda2

/dev/xvda2:
 Timing cached reads:   18670 MB in  1.99 seconds = 9389.01 MB/sec
 Timing buffered disk reads: 216 MB in  3.03 seconds =  71.40 MB/sec
```

28.7. VERIFYING THE RED HAT PROCESS AUTOMATION MANAGER CLUSTER

After configuring the cluster for Red Hat Process Automation Manager, create an asset to verify that the installation is working.

Procedure

1. In a web browser, enter **<node-IP-address>:8080/business-central**. Replace **<node-IP-address>** with the IP address of a particular node.

2. Enter the **admin** user credentials that you created during installation. The Business Central home page appears.

3. Select **Menu → Design → Projects**.

4. Open the **MySpace** space.

5. Click **Try Samples → Mortgages → OK**. The **Assets** window appears.
6. Click **Add Asset → Data Object**.
7. Enter **MyDataObject** in the **Data Object** field and click **OK**.
8. Click **Spaces → MySpace → Mortgages** and confirm that **MyDataObject** is in the list of assets.
9. Enter the following URL in a web browser, where **<node_IP_address>** is the address of a different node of the cluster:
   http://<node_IP_address>:8080/business-central
10. Enter the same credentials that you used to log in to Business Central on the first node, where you created the **MyDataObject** asset.
11. Select **Menu→ Design → Projects**.
12. Open the **MySpace** space.
13. Select the **Mortgages** project.
14. Verify that **MyDataObject** is in the asset list.
15. Delete the **Mortgages** project.

CHAPTER 29. KIE SERVER CLUSTERS IN A RUNTIME ENVIRONMENT

The primary benefit of clustering a KIE Server runtime environment is load balancing. If activity on one node of the cluster increases, that activity can be shared among the other nodes of the cluster to improve performance.

To create a KIE Server clustered runtime environment, you download and extract Red Hat JBoss EAP 7.4 and KIE Server. Then, you configure Red Hat JBoss EAP 7.4 for a cluster, start the cluster, and install KIE Server on each cluster node.

Optionally, you can then cluster the headless Process Automation Manager controller.

NOTE

Do not connect KIE Server to Business Central in high availability (HA) on premise environments.

Business Central instances are not able to keep in sync with the status of each KIE Server. For example, if a KIE Server is up but not in sync, Business Central will not be able to deploy through that instance.

29.1. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.4 AND KIE SERVER

Complete the steps in this section to download and install Red Hat JBoss EAP 7.4 and KIE Server for installation in a clustered environment.

Procedure

1. Download Red Hat JBoss EAP 7.4 on each node of the cluster:
   a. 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: Enterprise Application Platform**
      - **Version: 7.4**
   b. Click **Download** next to **Red Hat JBoss Enterprise Application Platform 7.4.14** (**jboss-eap-7.4.14.zip**).
2. Extract the **jboss-eap-7.4.14.zip** file. The **jboss-eap-7.4/jboss-eap-7.4** directory is referred to as ***EAP_HOME***.
3. Download and apply the latest Red Hat JBoss EAP patch, if available.
4. Download KIE Server:
   a. Navigate to the Software Downloads page in the Red Hat Customer Portal and select the product and version from the drop-down options:
      - **PRODUCT:** Process Automation Manager
      - **Version:** 7.13.5

b. Download **Red Hat Process Automation Manager 7.13.5 KIE Server for All Supported EE8 Containers** (**rhpam-7.13.5-kie-server-ee8.zip**).

5. Extract the **rhpam-7.13.5-kie-server-ee8.zip** archive to a temporary directory. In the following examples this directory is called ***TEMP_DIR***.

6. Copy the ***TEMP_DIR*/rhpam-7.13.5-kie-server-ee8/rhpam-7.13.5-kie-server-ee8/kie-server.war** directory to ***EAP_HOME*/standalone/deployments/**.

WARNING

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

7. Copy the contents of the ***TEMP_DIR*/rhpam-7.13.5-kie-server-ee8/rhpam-7.13.5-kie-server-ee8/SecurityPolicy/** to ***EAP_HOME*/bin**.

8. When prompted to overwrite files, click **Replace**.

9. In the ***EAP_HOME*/standalone/deployments/** directory, create an empty file named **kie-server.war.dodeploy**. This file ensures that KIE Server is automatically deployed when the server starts.

10. Download and apply the latest Red Hat Decision Manager patch, if available.

11. Navigate to the ***EAP_HOME*/bin** directory.

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

```
$ ./bin/jboss-cli.sh --commands="embed-server --std-
out=echo,/subsystem=elytron/filesystem-realm=ApplicationRealm:add-identity(identity=
<USERNAME>),/subsystem=elytron/filesystem-realm=ApplicationRealm:set-
password(identity=<USERNAME>, clear=
{password='<PASSWORD>'}),/subsystem=elytron/filesystem-realm=ApplicationRealm:add-
identity-attribute(identity=<USERNAME>, name=role, value=['kie-server'])"
```

13. Make a note of your user names and passwords.

14. To start the cluster, navigate to ***EAP_HOME*/bin** and 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
```

29.2. CLUSTERING KIE SERVERS WITH THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

The Process Automation Manager controller is integrated with Business Central. However, if you do not install Business Central, you can install the headless Process Automation Manager controller and use the REST API or the KIE Server Java Client API to interact with it.

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.4 or later 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.
- An NFS server with a shared folder is available as described in Section 28.3, "Configuring an NFS version 4 server".

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:** 7.13.5
2. Download **Red Hat Process Automation Manager 7.13.5 Add Ons**(the **rhpam-7.13.5-add-ons.zip** file).
3. Extract the **rhpam-7.13.5-add-ons.zip** file. The **rhpam-7.13.5-controller-ee7.zip** file is in the extracted directory.
4. Extract the **rhpam-7.13.5-controller-ee7.zip** archive to a temporary directory. In the following examples this directory is called ***TEMP_DIR***.
5. Copy the ***TEMP_DIR*/rhpam-7.13.5-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-7.13.5-controller-ee7/SecurityPolicy/** directory to ***EAP_HOME*/bin**.
7. When prompted to overwrite files, click **Yes**.

8. 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.

9. Open the ***EAP_HOME*/standalone/configuration/standalone.xml** file in a text editor.

10. Add the following properties to the **<system-properties>** element and replace **<NFS_STORAGE>** with the absolute path to the NFS storage where the template configuration is stored:

```
<system-properties>
  <property name="org.kie.server.controller.templatefile.watcher.enabled" value="true"/>
  <property name="org.kie.server.controller.templatefile" value="<NFS_STORAGE>"/>
</system-properties>
```

Template files contain default configurations for specific deployment scenarios.

If the value of the **org.kie.server.controller.templatefile.watcher.enabled** property is set to true, a separate thread is started to watch for modifications of the template file. The default interval for these checks is 30000 milliseconds and can be further controlled by the **org.kie.server.controller.templatefile.watcher.interval** system property. If the value of this property is set to false, changes to the template file are detected only when the server restarts.

11. To start the headless Process Automation Manager controller, navigate to ***EAP_HOME*/bin** and enter the following command:

- On Linux or UNIX-based systems:

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

- On Windows:

```
standalone.bat
```

CHAPTER 30. ADDITIONAL RESOURCES

- *Installing and configuring Red Hat Decision Manager on Red Hat JBoss EAP 7.4*
- *Planning a Red Hat Decision Manager installation*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 4 using Operators*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 3 using templates*

PART IV. INSTALLING AND CONFIGURING RED HAT PROCESS AUTOMATION MANAGER ON RED HAT JBOSS WEB SERVER

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. Starting with this release, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files. There are no Decision Manager artifacts. The Red Hat Decision Manager subscription, support entitlements, and fees remain the same. Red Hat Decision Manager subscribers will continue to receive full support for the decision management and optimization capabilities of Red Hat Decision Manager. The business process management (BPM) capabilities of Red Hat Process Automation Manager are exclusive to Red Hat Process Automation Manager subscribers. They are available for use by Red Hat Decision Manager subscribers but with development support services only. Red Hat Decision Manager subscribers can upgrade to a full Red Hat Process Automation Manager subscription at any time to receive full support for BPM features.

This document describes how to install Red Hat Process Automation Manager 7.13 on JBoss Web Server.

NOTE

Support for Red Hat Decision Manager on Red Hat JBoss Web Server and Apache Tomcat is now in the maintenance phase. Red Hat will continue to support Red Hat Process Automation Manager on these platforms with the following limitations:

- Red Hat will not release new certifications or software functionality.
- Red Hat will release only qualified security patches that have a critical impact and mission-critical bug fix patches.

In the future, Red Hat might direct customers to migrate to new platforms and product components that are compatible with the Red Hat hybrid cloud strategy.

Prerequisites

- You have reviewed the information in *Planning a Red Hat Decision Manager installation* .
- You have installed Red Hat JBoss Web Server 5.5.1. For information about installing Red Hat JBoss Web Server, see the *Red Hat JBoss Web Server Installation Guide* .

CHAPTER 31. ABOUT RED HAT DECISION MANAGER

Red Hat Decision Manager is an open source decision management platform that combines business rules management, complex event processing, Decision Model & Notation (DMN) execution, and Red Hat build of OptaPlanner for solving planning problems. It automates business decisions and makes that logic available to the entire business.

Business assets such as rules, decision tables, and DMN models are organized in projects and stored in the Business Central repository. This ensures consistency, transparency, and the ability to audit across the business. Business users can modify business logic without requiring assistance from IT personnel.

The product is made up of Business Central and KIE Server.

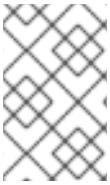
- KIE Server is the server where rules and other artifacts are executed. It is used to instantiate and execute rules and solve planning problems. KIE Server provides the runtime environment for business assets and accesses the data stored in the assets repository (knowledge store).
- Business Central is the graphical user interface where you create and manage business rules that KIE Server executes.

Business Central is also available as a standalone JAR file. You can use the Business Central standalone JAR file to run Business Central without needing to deploy it to an application server.

Red Hat JBoss Web Server is an enterprise ready web server designed for medium and large applications, based on Tomcat. Red Hat JBoss Web Server provides organizations with a single deployment platform for Java Server Pages (JSP) and Java Servlet technologies, PHP, and CGI.

On a Red Hat JBoss Web Server installation, you can install KIE Server and the headless Process Automation Manager controller. Alternatively, you can run the standalone Business Central JAR file.

The instructions in this document explain how to install Red Hat Process Automation Manager in a Red Hat JBoss Web Server instance.



NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

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

- *Installing and configuring Red Hat Decision Manager on Red Hat JBoss EAP 7.4*
- *Installing and configuring KIE Server on IBM WebSphere Application Server*
- *Installing and configuring KIE Server on Oracle WebLogic Server*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 4 using Operators*
- *Deploying an Red Hat Decision Manager environment on Red Hat OpenShift Container Platform 3 using templates*

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

31.1. RED HAT DECISION MANAGER COMPONENTS

The product is made up of Business Central and KIE Server.

- Business Central is the graphical user interface where you create and manage business rules. You can install Business Central in a Red Hat JBoss EAP instance or on the Red Hat OpenShift Container Platform (OpenShift).
  Business Central is also available as a standalone JAR file. You can use the Business Central standalone JAR file to run Business Central without deploying it to an application server.

- KIE Server is the server where rules and other artifacts are executed. It is used to instantiate and execute rules and solve planning problems. You can install KIE Server in a Red Hat JBoss EAP instance, in a Red Hat JBoss EAP cluster, on OpenShift, in an Oracle WebLogic server instance, in an IBM WebSphere Application Server instance, or as a part of Spring Boot application.
  You can configure KIE Server to run in managed or unmanaged mode. If KIE Server is unmanaged, you must manually create and maintain KIE containers (deployment units). A KIE container is a specific version of a project. If KIE Server is managed, the Process Automation Manager controller manages the KIE Server configuration and you interact with the Process Automation Manager controller to create and maintain KIE containers.

  On a Red Hat JBoss Web Server installation, you can install KIE Server and the headless Process Automation Manager controller. Alternatively, you can run the standalone Business Central JAR file.

31.2. RED HAT DECISION MANAGER ROLES AND USERS

To access Business Central or KIE Server, you must create users and assign them appropriate roles before the servers are started. You can create users and roles when you install Business Central or KIE Server.

If both Business Central and KIE Server are running on a single instance, a user who is authenticated for Business Central can also access KIE Server.

However, if Business Central and KIE Server are running on different instances, a user who is authenticated for Business Central must be authenticated separately to access KIE Server. For example, if a user who is authenticated on Business Central but not authenticated on KIE Server tries to view or manage process definitions in Business Central, a 401 error is logged in the log file and the **Invalid credentials to load data from remote server. Contact your system administrator.** message appears in Business Central.

This section describes Red Hat Decision Manager user roles.

NOTE

The **admin**, **analyst**, and **rest-all** roles are reserved for Business Central. The **kie-server** role is reserved for KIE Server. For this reason, the available roles can differ depending on whether Business Central, KIE 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 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 Decision Manager.

- **analyst**: Users with the **analyst** role have access to all high-level features. They can model 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.
- **rest-all**: Users with the **rest-all** role can access Business Central REST capabilities.
- **kie-server**: Users with the **kie-server** role can access KIE Server REST capabilities.

CHAPTER 32. DOWNLOADING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLATION FILES

You can use the installer JAR file or deployable ZIP files to install Red Hat Process Automation Manager. You can run the installer in interactive or command line interface (CLI) mode. Alternatively, you can extract and configure the Business Central and KIE Server deployable ZIP files. If you want to run Business Central without deploying it to an application server, download the Business Central Standalone JAR file.

Download a Red Hat Process Automation Manager distribution that meets your environment and installation requirements.

NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

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:** 7.13.5
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 Web Server, download **Red Hat Process Automation Manager 7.13.5 Installer** (**rhpam-installer-7.13.5.jar**). The installer graphical user interface guides you through the installation process.
   - To install KIE Server on Red Hat JBoss Web Server using the deployable ZIP file, download the **Red Hat Process Automation Manager 7.13.5 Add Ons**(**rhpam-7.13.5-add-ons.zip**) file.
     The ZIP file does not require a graphical user interface.
   - To run Business Central without needing to deploy it to an application server, download **Red Hat Process Automation Manager 7.13.5 Business Central Standalone** (**rhpam-7.13.5-business-central-standalone.jar**).

CHAPTER 33. USING THE RED HAT PROCESS AUTOMATION MANAGER INSTALLER

This section describes how to install Business Central and KIE Server 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 Web Server 5.5.1 server installation. You can run the installer in interactive or command line interface (CLI) mode.

Next steps:

Follow the instructions in one of the following sections:

- Section 33.1, “Using the installer in interactive mode”
- Section 33.2, “Using the installer in CLI mode”

33.1. 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 Web Server 5.5.1 server installation.

NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

NOTE

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

Prerequisites

- The **Red Hat Process Automation Manager 7.13.5 Installer**has been downloaded. For instructions, see Chapter 32, *Downloading the Red Hat Process Automation Manager installation files*.
- 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 Web Server 5.5.1 server installation is available.
- Sufficient user permissions to complete the installation are granted.

  NOTE

  Ensure that you are logged in with a user that has write permission for Tomcat.

- 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. For a list of system properties, see Appendix B, *Business Central system properties*.

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-7.13.5.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-7.13.5.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-7.13.5.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 Web Server 5.5.1 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**.
   You cannot install Business Central on Red Hat JBoss Web Server. You can only install it on Red Hat JBoss EAP. However, you can install KIE Server and the headless Process Automation Manager controller on Red Hat JBoss Web Server. The headless Process Automation Manager controller is used to manage KIE Server. Install the headless Process Automation Manager controller if you plan to manage multiple KIE Server instances.

5. Create a user and click **Next**. By default, if you install both Business Central and KIE Server in the same container the new user is given the **admin**, **kie-server**, and **rest-all** roles. If you install only KIE Server, the user is given the **kie-server** role. The **kie-server** role is required to access KIE Server REST capabilities.

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 KIE Server.

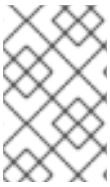
6. On the **Installation Overview** page, review the components that you will install and click **Next** to start the installation.
7. When the installation has completed, click **Next**.
8. When **Processing finished** appears at the top of the screen, click **Next** to complete the installation.
9. Optional: 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 to repeat the Red Hat Process Automation Manager installation on multiple systems with the same type of server and the same configuration as the original installation. If necessary, update the **installpath** parameter in the **auto.xml** file. To perform an installation using the XML file, enter the following command:

```
java -jar rhpam-installer-7.13.5.jar <path-to-auto.xml-file>
```

You have successfully installed Red Hat Process Automation Manager using the installer.

33.2. USING THE INSTALLER IN CLI MODE

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



NOTE

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. You must install Red Hat Process Automation Manager in order to use Red Hat Decision Manager.

NOTE

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

Prerequisites

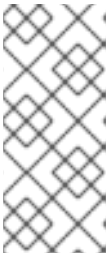
- The **Red Hat Process Automation Manager 7.13.5 Installer** has been downloaded. For instructions, see Chapter 32, *Downloading the Red Hat Process Automation Manager installation files*.

- 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 Web Server 5.5.1 server installation is available.
- Sufficient user permissions to complete the installation are granted.

NOTE

Ensure that you are logged in with a user that has write permission for Tomcat.

- 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. For a list of system properties, see Appendix B, *Business Central system properties*.

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-7.13.5.jar -console
```

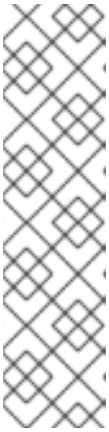
The command-line interactive process will start and display the End-User License Agreement.

```
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 or Red Hat
JBoss Web Server 5. For more information, see
https://access.redhat.com/articles/3405381[Red Hat Process Automation Manager 7
Supported Configurations].
```

3. Enter the parent directory of an existing Red Hat JBoss Web Server 5.5.1 installation. The installer will verify the location of the installation at the location provided. Enter **1** to confirm and continue.
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 KIE 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. To perform an installation using the XML file, enter the following command:

```
java -jar rhpam-installer-7.13.5.jar <path-to-auto.xml-file>
```

8. If you installed only KIE Server, repeat these steps to install the headless Process Automation Manager controller on a separate server.

CHAPTER 34. KIE SERVER ZIP FILE INSTALLATION AND CONFIGURATION

You can install KIE Server using the **rhpam-7.13.5-kie-server-jws.zip** file available from the **Red Hat Process Automation Manager 7.13.5 Add Ons** (**rhpam-7.13.5-add-ons.zip**) file on the Customer Portal .

34.1. INSTALLING KIE SERVER FROM ZIP FILES

KIE Server provides the runtime environment for business assets and accesses the data stored in the assets repository (knowledge store). You can use ZIP files to install KIE Server on an existing Red Hat JBoss Web Server 5.5.1 server instance.

NOTE

To use the installer JAR file to install KIE Server, see Chapter 33, *Using the Red Hat Process Automation Manager installer*.

- The **Red Hat Process Automation Manager 7.13.5 Add Ons**(**rhpam-7.13.5-add-ons.zip**) file has been downloaded, as described in Chapter 32, *Downloading the Red Hat Process Automation Manager installation files*.
- A backed-up Red Hat JBoss Web Server 5.5.1 server installation is available. The base directory of the Red Hat JBoss Web Server installation is referred to as ***JWS_HOME***.
- Sufficient user permissions to complete the installation are granted.

Procedure

1. Extract the **rhpam-7.13.5-add-ons.zip** file.
2. From the extracted **rhpam-7.13.5-add-ons.zip** file, extract the following files:
   - **rhpam-7.13.5-kie-server-jws.zip**
   - **rhpam-7.13.5-process-engine.zip**

   In the following instructions, the directory that contains the extracted **rhpam-7.13.5-kie-server-jws.zip** file is called **JWS_TEMP_DIR** and the directory that contains the extracted **rhpam-7.13.5-process-engine.zip** file is called **ENGINE_TEMP_DIR**.
3. Copy the **JWS_TEMP_DIR/rhpam-7.13.5-kie-server-jws/kie-server.war** directory to the ***JWS_HOME*/tomcat/webapps** directory.

   NOTE

   Ensure the names of the Red Hat Decision Manager deployments you copy do not conflict with your existing deployments in the Red Hat JBoss Web Server instance.
4. Remove the **.war** extensions from the **kie-server.war** folder.
5. Move the **kie-tomcat-integration-7.67.0.Final-redhat-00024.jar** file from the **ENGINE_TEMP_DIR** directory to the ***JWS_HOME*/tomcat/lib** directory.

6. Move the **jboss-jacc-api-<VERSION>.jar**, **slf4j-api-<VERSION>.jar**, and **slf4j-jdk14-<VERSION>.jar** files from the **ENGINE_TEMP_DIR/lib** directory to the ***JWS_HOME*/tomcat/lib** directory, where **<VERSION>** is the version artifact file name, in the **lib** directory.

7. Add the following line to the **<host>** element in the ***JWS_HOME*/tomcat/conf/server.xml** file after the last Valve definition:

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

8. Open the ***JWS_HOME*/tomcat/conf/tomcat-users.xml** file in a text editor.

9. Add users and roles to the ***JWS_HOME*/tomcat/conf/tomcat-users.xml** file. In the following example, **<ROLE_NAME>** is a role supported by Red Hat Decision Manager. **<USER_NAME>** and **<USER_PWD>** are the user name and password of your choice:

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

   If a user has more than one role, as shown in the following example, separate the roles with a comma:

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

10. Complete one of the following steps in the ***JWS_HOME*/tomcat/bin** directory:

    - On Linux or UNIX, create the **setenv.sh** file with the following content:

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

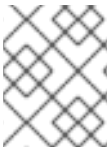
    - On Windows, add the following content to the **setenv.bat** file:

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

CHAPTER 35. SECURING PASSWORDS WITH A KEYSTORE

You can use a keystore to encrypt passwords that are used for communication between Business Central and KIE Server. You should encrypt both controller and KIE Server passwords. If Business Central and KIE 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 KIE Server is not configured with JCEKS, KIE Server passwords are stored in system properties in plain text form.

Prerequisites

- KIE Server is installed in Red Hat JBoss Web Server.
- Java 8 or higher is installed.

Procedure

1. Open the ***JWS_HOME*/tomcat/conf/tomcat-users.xml** file in a text editor.
2. Add a KIE Server user with the **kie-server** role to the ***JWS_HOME*/tomcat/conf/tomcat-users.xml** file. In the following example, replace **<USER_NAME>** and **<PASSWORD>** with the user name and password of your choice.

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

3. 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

4. When prompted, enter the password for the KIE Server user that you created.
5. To set the system properties, complete one of these steps in the ***JWS_HOME*/tomcat/bin** directory and replace the variables as described in the following table:

NOTE

If Business Central or the standalone controller are installed in separate instances from Red Hat JBoss Web Server, do not add the **kie.keystore.key.server.alias** and **kie.keystore.key.server.pwd** properties to **CATALINA_OPTS**.

- On Linux or UNIX, create the **setenv.sh** file with the following content:

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

- On Windows, add the following content to the **setenv.bat** file:

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

Table 35.1. System properties used to load a KIE 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_ALIAS>**	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_ALIAS>**	Alias of the key for default REST Process Automation Controller where the password is stored
kie.keystore.key.ctrl.pwd	**<KEY_CONTROL_PWD>**	Password of the alias for default REST Process Automation Controller with the stored password

6. Start KIE Server to verify the configuration.

CHAPTER 36. VERIFYING THE KIE SERVER INSTALLATION

Verify that KIE Server is installed correctly.

Prerequisites

- KIE Server is installed and configured.

Procedure

1. To start KIE Server, enter one of the following commands in the ***JWS_HOME*/tomcat/bin** directory:
   - On Linux or UNIX-based systems:

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

   - On Windows:

     ```
     startup.bat
     ```

2. After a few minutes, review the files in the ***JWS_HOME*/tomcat/logs** directory and correct any errors.
3. To verify that KIE Server is working correctly, enter **http://localhost:8080/kie-server/services/rest/server** in a web browser.
4. Enter the user name and password stored in the **tomcat-users.xml** file.

CHAPTER 37. DOWNLOADING AND INSTALLING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

You can configure KIE Server to run in managed or unmanaged mode. If KIE Server is unmanaged, you must manually create and maintain KIE containers (deployment units). If KIE Server is managed, the Process Automation Manager controller manages the KIE 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, use the **Execution Server** page to create and maintain KIE containers. However, if you do not install Business Central, you can install the headless Process Automation Manager controller and use the REST API or the KIE Server Java Client API to interact with it.

Prerequisites

- The **Red Hat Process Automation Manager 7.13.5 Add Ons**(**rhpam-7.13.5-add-ons.zip**) file has been downloaded, as described in Chapter 32, *Downloading the Red Hat Process Automation Manager installation files*.
- A Red Hat JBoss Web Server 5.5.1 server installation is available. The base directory of the Red Hat JBoss Web Server installation is referred to as ***JWS_HOME***.
- Sufficient user permissions to complete the installation are granted.

Procedure

1. Extract the **rhpam-7.13.5-add-ons.zip** file. The **rhpam-7.13.5-controller-jws.zip** file is in the extracted directory.
2. Extract the **rhpam-7.13.5-controller-jws.zip** archive to a temporary directory. In the following examples this directory is called ***TEMP_DIR***.
3. Copy the ***TEMP_DIR*/rhpam-7.13.5-controller-jws.zip/controller.war** directory to the ***JWS_HOME*/tomcat/webapps** directory.

NOTE

Ensure the names of the Red Hat Decision Manager deployments you copy do not conflict with your existing deployments in the Red Hat JBoss Web Server instance.

4. Remove the **.war** extensions from the **controller.war** folder.
5. Copy the contents of the ***TEMP_DIR*/rhpam-7.13.5-controller-jws/SecurityPolicy/** directory to ***JWS_HOME*/bin**
6. When prompted to overwrite files, select **Yes**.
7. Add the **kie-server** role and user to the ***JWS_HOME*/tomcat/conf/tomcat-users.xml** file. In the following example, **<USER_NAME>** and **<PASSWORD>** are the user name and password of your choice:

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

8. Complete one of the following tasks in the ***JWS_HOME*/tomcat/bin** directory of the instance running KIE Server:

- On Linux or UNIX, create the **setenv.sh** file with the following content:

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

- On Windows, add the following content to the **setenv.bat** file:

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

In the preceding examples, replace the following variables:

- Replace **<CONTROLLER_USER>** and **<CONTROLLER_PWD>** with the user name and password for the **kie-server** role that you defined earlier in this procedure.
- Replace **<KIE_SERVER_ID>** with a unique identifier.
- Replace **<CONTROLLER_HOST>:<CONTROLLER_PORT>** with the IP address (host and port) of the controller. If you use the same server for KIE Server and the controller, **<CONTROLLER_HOST>:<CONTROLLER_PORT>** is **localhost:8080**.

9. In the ***JWS_HOME*/tomcat/bin** directory of the instance running the headless Process Automation Manager controller, create a readable **setenv.sh** file with the following content, where **<USERNAME>** is the KIE Server user and **<USER_PWD>** is the password for that user: **CATALINA_OPTS="-Dorg.kie.server.user=<USERNAME> -Dorg.kie.server.pwd= <USER_PWD>"**

10. To start the headless Process Automation Manager controller, enter one of the following commands in the ***JWS_HOME*/tomcat/bin** directory:

- On Linux or UNIX-based systems:

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

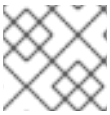
- On Windows:

```
startup.bat
```

11. After a few minutes, review the ***JWS_HOME*/tomcat/logs** directory and correct any errors.
12. To verify that the headless Process Automation Manager controller is working correctly, enter **http://<CONTROLLER_HOST>:<CONTROLLER_PORT>/controller/rest/controller/management/servers** in a web browser. If you use the same server for KIE Server and the controller, **<CONTROLLER_HOST>:<CONTROLLER_PORT>** is **localhost:8080**.
13. Enter the user name and password stored in the **tomcat-users.xml** file.

CHAPTER 38. CONFIGURING AND RUNNING STANDALONE BUSINESS CENTRAL

You can use the Business Central standalone JAR file to run Business Central without deploying it to an application server. You can use sample configuration files to run the Business Central standalone JAR file out of the box or you can customize the sampfiles for your requirements.



NOTE

This JAR file is supported only when it is run on Red Hat Enterprise Linux.

Prerequisites

- The **Red Hat Process Automation Manager 7.13.5 Business Central Standalone**(**rhpam-7.13.5-business-central-standalone.jar**) and the **Red Hat Process Automation Manager 7.13.5 Add Ons** (**rhpam-7.13.5-add-ons.zip**) files have been downloaded from the Software Downloads page for Red Hat Process Automation Manager 7.13, as described in Chapter 32, *Downloading the Red Hat Process Automation Manager installation files* .

Procedure

1. Extract the downloaded **rhpam-7.13.5-addons.zip** to a temporary directory. This archive includes the **rhpam-7.13.5-standalone-sample-configuration.zip** file.
2. Extract the **rhpam-7.13.5-standalone-sample-configuration.zip** file to the directory that contains the **rhpam-7.13.5-business-central-standalone.jar** file. The **rhpam-7.13.5-standalone-sample-configuration.zip** file contains the following sample configuration files:
   - **application-script.cli**: Sample script for adding a user and kie server system properties
   - **kie-fs-realm-users**: Sample user data
     You can run the **rhpam-7.13.5-business-central-standalone.jar** files with the sample data provided in the configuration files or you can customize the data for your requirements.
3. To customize the configuration data, complete the following steps:
   a. Edit the **application-script.cli** file to include an administrative user with **admin**, **user**, **rest-all**, **rest-client** and **kie-server** roles. In the following example, replace **<USERNAME>** and **<PASSWORD>** with your username and password of the user you want to create.

```
/subsystem=elytron/filesystem-realm=KieRealm:add-identity(identity=<USERNAME>)
/subsystem=elytron/filesystem-realm=KieRealm:set-password(identity=<USERNAME>,
clear={password="<PASSWORD>"})
/subsystem=elytron/filesystem-realm=KieRealm:add-identity-attribute(identity=
<USERNAME>, name=role, value=["admin","user","rest-all","rest-client","kie-server"])
```

4. To run the Business Central standalone JAR file, enter the following command:

```
java -jar rhpam-7.13.5-business-central-standalone.jar --cli-script=application-script.cli
```

5. To set application properties when you run the JAR file, include the **-D<PROPERTY>=<VALUE>** parameter in the command, where **<PROPERTY>** is the name of a supported application property and **<VALUE>** is the property value:

```
java -jar rhpam-7.13.5-business-central-standalone.jar --cli-script=application-script.cli -
D<PROPERTY>=<VALUE> -D<PROPERTY>=<VALUE>
```

For example, to run Business Central and connect to KIE Server as the user **controllerUser**, enter:

```
java -jar rhpam-7.13.5-business-central-standalone.jar \
 --cli-script=application-script.cli \
-Dorg.kie.server.user=controllerUser \
-Dorg.kie.server.pwd=controllerUser1234
```

Doing this enables you to deploy containers to KIE Server. See Appendix A, *Business Central system properties* for more information.

NOTE

To enable user and group management in Business Central, set the value of the **org.uberfire.ext.security.management.wildfly.cli.folderPath** property to **kie-fs-realm-users**.

CHAPTER 39. MAVEN SETTINGS AND REPOSITORIES FOR RED HAT DECISION MANAGER

When you create a Red Hat Decision 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 Decision Manager projects to retrieve dependencies from the public Red Hat Decision Manager repository by modifying the Maven project object model (POM) file (**pom.xml**). You can also configure Business Central and KIE Server to use an external Maven repository or prepare a Maven mirror for offline use.

For more information about Red Hat Decision Manager packaging and deployment options, see *Packaging and deploying an Red Hat Decision Manager project* .

39.1. CONFIGURING MAVEN USING THE PROJECT CONFIGURATION FILE (POM.XML)

To use Maven for building and managing your Red Hat Decision Manager projects, you must create and configure the POM file (**pom.xml**). This file holds configuration information for your project. For more information, see Apache Maven Project.

Procedure

1. Generate a Maven project. A **pom.xml** file is automatically generated when you create a Maven project.

2. Edit the **pom.xml** file to add more dependencies and new repositories.
   Maven downloads all of the JAR files and the dependent JAR files from the Maven repository when you compile and package your project.

Find the schema for the **pom.xml** file at http://maven.apache.org/maven-v4_0_0.xsd. For more information about POM files, see Apache Maven Project POM .

39.2. MODIFYING THE MAVEN SETTINGS FILE

Red Hat Decision Manager uses Maven **settings.xml** file to configure it's Maven execution. You must create and activate a profile in the **settings.xml** file and declare the Maven repositories used by your Red Hat Decision Manager projects.

For information about the Maven **settings.xml** file, see the Apache Maven Project Setting Reference.

Procedure

1. In the **settings.xml** file, declare the repositories that your Red Hat Decision Manager projects use. Usually, this is either the online Red Hat Decision Manager Maven repository or the Red Hat Decision Manager Maven repository that you download from the Red Hat Customer Portal and any repositories for custom artifacts that you want to use.

2. Ensure that Business Central or KIE Server is configured to use the **settings.xml** file. For example, specify the **kie.maven.settings.custom=<SETTINGS_FILE_PATH>** property where **<SETTINGS_FILE_PATH>** is the path to the **settings.xml** file.
   On Red Hat JBoss Web Server, for KIE Server add **-Dkie.maven.settings.custom=<SETTINGS_FILE_PATH>** to the **CATALINA_OPTS** section of the **setenv.sh** (Linux) or **setenv.bat** (Windows) file. For standalone Business Central, enter the following command:

```
java -jar rhpam-7.13.5-business-central-standalone.jar --cli-script=application-script.cli -
Dkie.maven.settings.custom=<SETTINGS_FILE_PATH>
```

39.3. ADDING MAVEN DEPENDENCIES FOR RED HAT DECISION MANAGER

To use the correct Maven dependencies in your Red Hat Decision 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>7.13.5.redhat-00002</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 dependencies are provided automatically by Business Central.

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

   - For a basic Red Hat Decision 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 KIE Server, declare the following dependencies:

  Client application KIE Server dependencies

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

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

  Client dependency

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

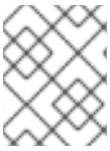
- When creating a JAR file that includes assets, such as rules and process definitions, specify the packaging type for your Maven project as **kjar** and use **org.kie:kie-maven-plugin** to process the **kjar** packaging type located under the **<project>** element. In the following example, **${kie.version}**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

39.4. 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:
2. 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:** 7.13.5
     a. Download and extract the **Red Hat Process Automation Manager 7.13.5 Offliner Content List** (**rhpam-7.13.5-offliner.zip**) product deliverable file.
     b. Extract the contents of the **rhpam-7.13.5-offliner.zip** file into any directory.
     c. Change to the directory and enter the following command:

     ```
     ./offline-repo-builder.sh offliner.txt
     ```

     This command creates the **repository** subdirectory and downloads the necessary artifacts into this subdirectory. This is the mirror repository.

     If a message reports that some downloads have failed, run the same command again. If downloads fail again, contact Red Hat support.
3. If you developed services outside of 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 path of the **pom.xml** file of the project.

d. Copy the contents of the local Maven cache directory (~/**.m2/repository**) to the **repository** subdirectory that was created.

4. Copy the contents of the **repository** subdirectory to a directory on the computer on which you deployed Red Hat Process Automation Manager. This directory becomes the offline Maven mirror repository.

5. Create and configure a **settings.xml** file for your Red Hat Process Automation Manager deployment as described in Section 39.2, "Modifying the Maven settings file".

6. Make the following changes in the **settings.xml** file:

- Under the **<profile>** tag, if a **<repositories>** or **<pluginRepositores>** tag is missing, add the missing tags.

- Under **<repositories>** add the following content:

```
<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 content:

```
<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.

CHAPTER 40. 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 Decision 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. To import a project, do one of the following:
   - Click **Import Project**.
   - Select **Import Project** from the drop-down list.
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 41. INTEGRATING LDAP AND SSL

With Red Hat Process Automation Manager you can integrate LDAP and SSL through Red Hat Single Sign-On. For more information, see the *Red Hat Single Sign-On Server Administration Guide* .

APPENDIX B. BUSINESS CENTRAL SYSTEM PROPERTIES

The Business Central system properties listed in this section are passed to **standalone*.xml** files.

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**.

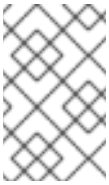
NOTE

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 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 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.

KIE Server nodes and Process Automation Manager controller

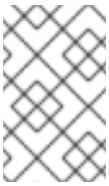
Use the following properties to configure the connections with the KIE Server nodes from the Process Automation Manager controller:

- **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 KIE 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 KIE 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.builder.cache.size**: Defines the cache size of the project builder. Default value: **20**.
- **org.kie.library.assets_per_page**: You can customize the number of assets per page in the project screen. Default value: **15**.
- **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

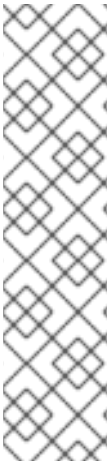
Use the following properties to switch between Business Central and KIE Server rendered forms:

- **org.jbpm.wb.forms.renderer.ext**: Switches the form rendering between Business Central and KIE 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 KIE Server rendered forms. Default value: **workbench**.

PART V. INSTALLING AND CONFIGURING KIE SERVER ON IBM WEBSPHERE APPLICATION SERVER

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. Starting with this release, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files. There are no Decision Manager artifacts. The Red Hat Decision Manager subscription, support entitlements, and fees remain the same. Red Hat Decision Manager subscribers will continue to receive full support for the decision management and optimization capabilities of Red Hat Decision Manager. The business process management (BPM) capabilities of Red Hat Process Automation Manager are exclusive to Red Hat Process Automation Manager subscribers. They are available for use by Red Hat Decision Manager subscribers but with development support services only. Red Hat Decision Manager subscribers can upgrade to a full Red Hat Process Automation Manager subscription at any time to receive full support for BPM features.

As a system administrator, you can configure your IBM WebSphere Application Server for Red Hat KIE Server and install KIE Server on that IBM server instance.



NOTE

Support for Red Hat Decision Manager on IBM WebSphere Application Server is now in the maintenance phase. Red Hat will continue to support Red Hat Process Automation Manager on IBM WebSphere Application Server with the following limitations:

- Red Hat will not release new certifications or software functionality.
- Red Hat will release only qualified security patches that have a critical impact and mission-critical bug fix patches.

In the future, Red Hat might direct customers to migrate to new platforms and product components that are compatible with the Red Hat hybrid cloud strategy.

Prerequisites

- An IBM WebSphere Application Server instance version 9.0 or later is installed. For complete installation instructions, see the IBM WebSphere Application Server product page .
- You have access to the WebSphere Integrated Solutions Console, usually at **http://<HOST>:9060/ibm/console**.

CHAPTER 42. KIE SERVER

KIE Server is the server where the rules and other artifacts for Red Hat Process Automation Manager are stored and run. KIE Server is a standalone built-in component that can be used to instantiate and execute rules through interfaces available for REST, Java Message Service (JMS), or Java client-side applications, and Red Hat build of OptaPlanner functionality through solvers.

Created as a web deployable WAR file, KIE Server can be deployed on any web container. The current version of KIE Server is included with default extensions for both Red Hat Decision Manager and Red Hat Process Automation Manager.

KIE Server has a low footprint with minimal memory consumption and therefore can be deployed easily on a cloud instance. Each instance of this server can open and instantiate multiple containers, which enables you to execute multiple rule services in parallel.

KIE Server can be integrated with other application servers, such as Oracle WebLogic Server or IBM WebSphere Application Server, to streamline Red Hat Process Automation Manager application management.

CHAPTER 43. IBM WEBSPHERE APPLICATION SERVER

IBM WebSphere Application Server is a flexible and secure web application server that hosts Java-based web applications and provides Java EE-certified run time environments. IBM WebSphere 9.0 supports Java SE 8 and is fully compliant with Java EE 7.

CHAPTER 44. INSTALLING AND RUNNING IBM WEBSPHERE APPLICATION SERVER

IBM WebSphere Application Server must be installed and running for you to apply many of the configurations that accommodate KIE Server. This section describes how to install and start IBM WebSphere.

For the most up-to-date and detailed installation instructions, see the IBM Knowledge Center.

Procedure

1. Download IBM Installation Manager version 1.8.5 or later from the IBM Installation Manager and Packaging Utility download links page. IBM Installation Manager is required for installing IBM WebSphere.
2. Extract the downloaded archive and run the following command as the root user in the new directory:

   ```
   sudo ./install
   ```

   The **IBM Installation Manager** opens.
3. Go to **File → Preferences** and click **Add Repository**.
4. In the **Add Repository** window, enter the repository URL for IBM WebSphere 9.0. You can find all the repository URLs in the Online product repositories for IBM WebSphere Application Server offerings page of the IBM Knowledge Center.
5. In your command terminal, navigate to the IBM WebSphere Application Server folder location that you specified during the installation.
6. Change to the **/bin** directory and run a command similar to the following example to create an IBM WebSphere profile, user name, and password. A profile defines the run time environment. The profile includes all the files that the server processes in the runtime environment and that you can change. The user is required for login.

   ```
   sudo ./manageprofiles.sh -create -profileName testprofile -profilePath /profiles/testprofile  -
   adminUserName websphere -adminPassword password123
   ```
7. In your command terminal, navigate to the **bin** directory within the profile that you created (for example, **/profiles/testprofile/bin**) and run the following command to start the IBM WebSphere Application Server instance:

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

   Replace **<SERVER_NAME>** with the IBM WebSphere Application Server name defined in **Servers → Server Types → IBM WebSphere Application Servers** of the WebSphere Integrated Solutions Console.
8. Open the following URL in a web browser:

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

   <HOST> is the system name or IP address of the target server.

For example, to start the WebSphere Integrated Solutions Console for a local instance of IBM WebSphere running on your system, enter the following URL in a web browser:

```
http://localhost:9060/ibm/console
```

9. When the login page of the WebSphere Integrated Solutions Console appears, enter your administrative credentials.

CHAPTER 45. CONFIGURING IBM WEBSPHERE APPLICATION SERVER FOR KIE SERVER

Before you deploy KIE Server with IBM WebSphere Application Server, you must configure system properties, security settings, JMS requirements, and other properties on IBM WebSphere. These configurations promote an optimal integration with KIE Server.

Prerequisites

- IBM WebSphere Application Server is installed and running.
- You are logged in to the WebSphere Integrated Solutions Console.

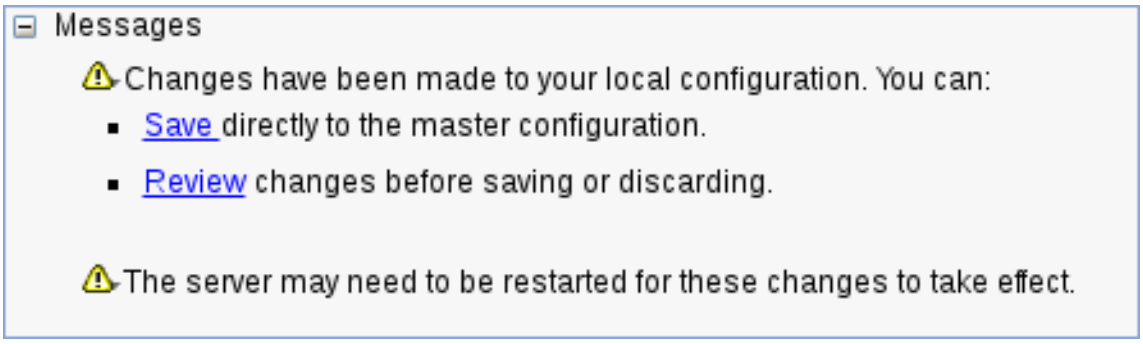
45.1. ENABLING ADMINISTRATIVE SECURITY

You must enable administrative security in the WebSphere Integrated Solutions Console so that you have the required permissions to create users and groups.

Procedure

1. In the WebSphere Integrated Solutions Console, click **Security → Global Security** and ensure that the option **Enable Application Security** is selected. This may already be selected and overridden at the server level.
2. Click **Security Configuration Wizard** and click **Next**.
3. Select the repository that contains the user information. For example, select **Federated repositories** for local configurations.
4. Click **Next**
5. Enter the **Primary administrative user name** and **Password**.
6. Click **Next** and then click **Finish**.
7. Click **Save** in the **Messages** window to save your changes to the primary configuration.

   Figure 45.1. Save security changes

8. In your command terminal, navigate to the IBM WebSphere Application Server **/bin** directory location that you specified during installation, and run the following commands to stop and restart IBM WebSphere to apply the security changes:

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

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

Replace **<SERVER_NAME>** with the IBM WebSphere Application Server name defined in **Servers → Server Types → IBM WebSphere Application Servers** of the WebSphere Integrated Solutions Console.

45.2. CONFIGURING JAVA MESSAGE SERVICE (JMS)

The Java Message Service (JMS) is a Java API that KIE Server uses to exchange messages with other application servers such as Oracle WebLogic Server and IBM WebSphere Application Server. You must configure your application server to send and receive JMS messages through KIE Server to ensure collaboration between the two servers.

45.2.1. Create a service bus and add IBM WebSphere Application Server

You must create a service bus and add the IBM WebSphere Application Server as a member in order to use JMS.

Procedure

1. In the WebSphere Integrated Solutions Console, navigate to **Service Integration → Buses → New**.
2. Enter a new bus name and clear the **Bus Security** option.
3. Click **Next** and then **Finish** to create the service bus.
4. Select the service bus that you created.
5. Under **Topology**, select **Bus Members → Add**.
6. In the **Add a New Bus Member** wizard, choose the IBM WebSphere Application Server and the type of message store for persistence. You can also specify the properties of the message store.
7. Click **Finish** to add the new bus member.

45.2.2. Create JMS connection factories

To enable messaging with KIE Server, you must create certain JMS connection factories for sending and receiving messages.

Prerequisites

- You have created a service bus for IBM WebSphere Application Server.

Procedure

1. In the WebSphere Integrated Solutions Console, navigate to **Resources → JMS → Connection Factories**.
2. Select the correct scope and click **New**.
3. Select the **Default Messaging Provider** option and click **OK**.

4. For each of the following required connection factories, enter the name of the connection factory (for example, **KIE.SERVER.REQUEST**) and the JNDI name (for example, **jms/cf/KIE.SERVER.REQUEST**), and then select the service bus from the **Bus Name** drop-down list. Leave the default values for the remaining options.

5. Click **Apply** and then click **Save** to save the changes to the primary configuration, and repeat for each required factory.

45.2.2.1. JMS connection factories for KIE Server

The following table lists the required Java Message Service (JMS) connection factories that enable JMS messaging with KIE Server:

Table 45.1. Required JMS connection factories for KIE Server

Name	Default value	Used for
KIE.SERVER.REQUEST	**jms/cf/KIE.SERVER.REQUEST**	Sending all requests to KIE Server
KIE.SERVER.RESPONSE	**jms/cf/KIE.SERVER.RESPONSE**	Receiving all responses produced by KIE Server

45.2.3. Create JMS queues

JMS queues are the destination end points for point-to-point messaging. You must create certain JMS queues to enable JMS messaging with KIE Server.

Prerequisites

- You have created a service bus for IBM WebSphere Application Server.

Procedure

1. In the WebSphere Integrated Solutions Console, navigate to **Resources → JMS → Queues**.

2. Select the correct scope and click **New**.

3. Select the **Default Messaging Provider** option and click **OK**.

4. For each of the following required queues, enter the name of the queue (for example, **KIE.SERVER.REQUEST**) and the JNDI name (for example, **jms/KIE.SERVER.REQUEST**), and then select the service bus from the **Bus Name** drop-down list.

5. From the **Queue Name** drop-down list, select the **Create Service Integration Bus Destination**, enter a unique identifier, and select the bus member that you created previously.

6. Click **Apply** and then click **Save** to save the changes to the primary configuration, and repeat for each required queue.

45.2.3.1. JMS queues for KIE Server

The following table lists the required Java Message Service (JMS) queues that enable JMS messaging with KIE Server:

Table 45.2. Required JMS queues for KIE Server

Name	Default value	Used for
KIE.SERVER.REQUEST	**jms/KIE.SERVER.REQUEST**	Sending all requests to KIE Server
KIE.SERVER.RESPONSE	**jms/KIE.SERVER.RESPONSE**	Receiving all responses produced by KIE Server

45.2.4. Create JMS activation specifications

A JMS activation specification is required in order to bridge the queue and the message-driven bean that enables JMS.

Prerequisites

- You have created a service bus for IBM WebSphere Application Server.
- You have created JMS queues.

Procedure

1. In the WebSphere Integrated Solutions Console, navigate to **Resources → JMS → Activation Specifications**.
2. Select the correct scope and click **New**.
3. Select the **Default Messaging Provider** option and click **OK**.
4. For each of the following required activation specifications, enter the name of the activation specification (for example, **KIE.SERVER.REQUEST**) and the JNDI name (for example, **jms/activation/KIE.SERVER.REQUEST**), and then select the service bus from the **Bus Name** drop-down list.
5. From the **Destination Type** drop-down list, select **Queue** and enter the name of the corresponding queue as a **Destination lookup** (for example, **jms/KIE.SERVER.REQUEST**).
6. Click **Apply** and then click **Save** to save the changes to the primary configuration, and repeat for each required activation specification.

45.2.4.1. JMS activation specifications for KIE Server

The following table lists the required Java Message Service (JMS) activation specifications that enable JMS messaging with KIE Server:

Table 45.3. Required JMS activation specifications for KIE Server

Name	Default value	Used for
KIE.SERVER.REQUEST	**jms/activation/KIE.SERVER.REQUEST**	Sending all requests to KIE Server

Name	Default value	Used for
KIE.SERVER.RESPONSE	**jms/activation/KIE.SERVER.RESPONSE**	Receiving all responses produced by KIE Server

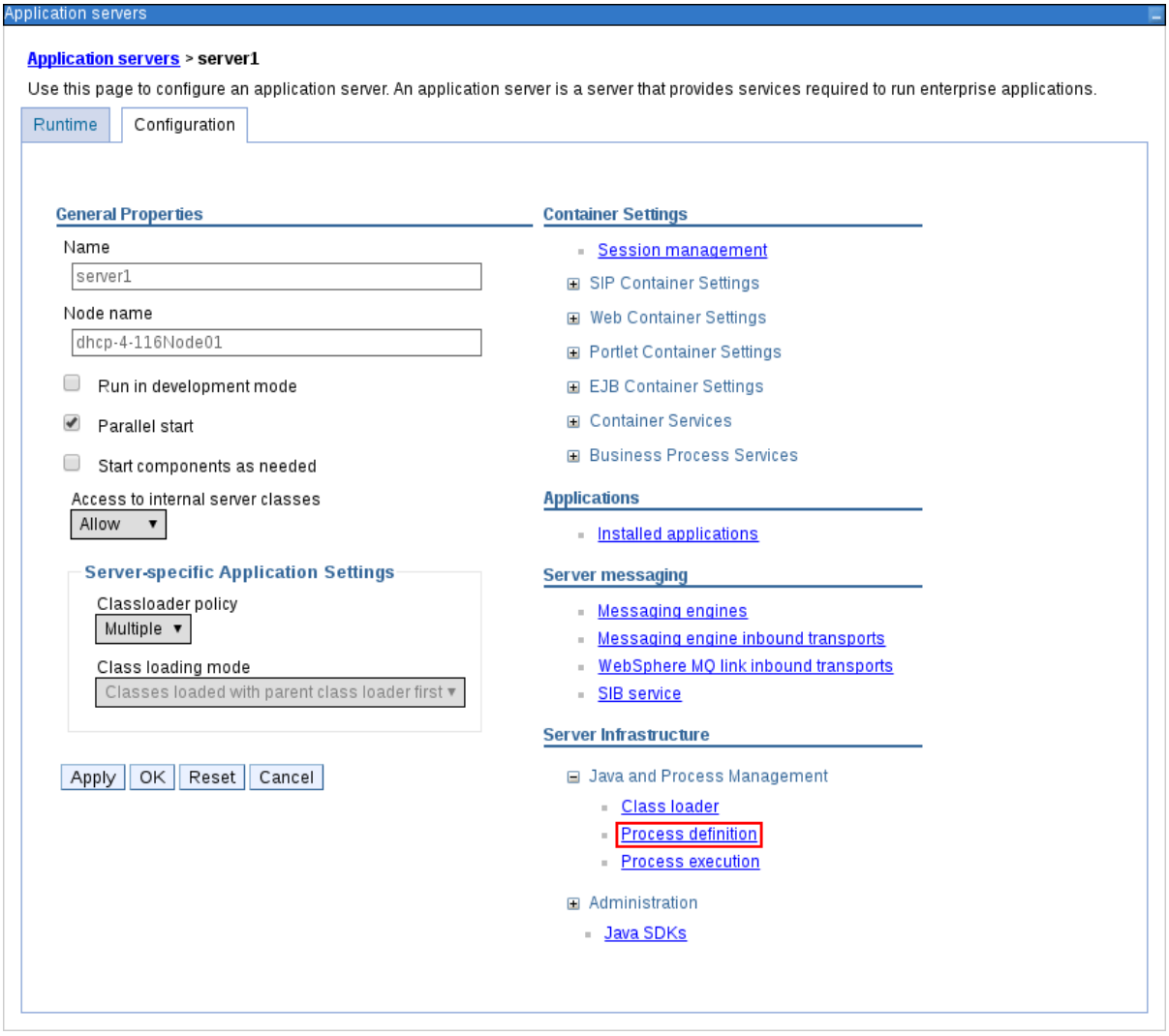
45.3. SETTING SYSTEM PROPERTIES IN IBM WEBSPHERE APPLICATION SERVER

Set the system properties listed in this section on your IBM WebSphere Application Server before you deploy KIE Server.

Procedure

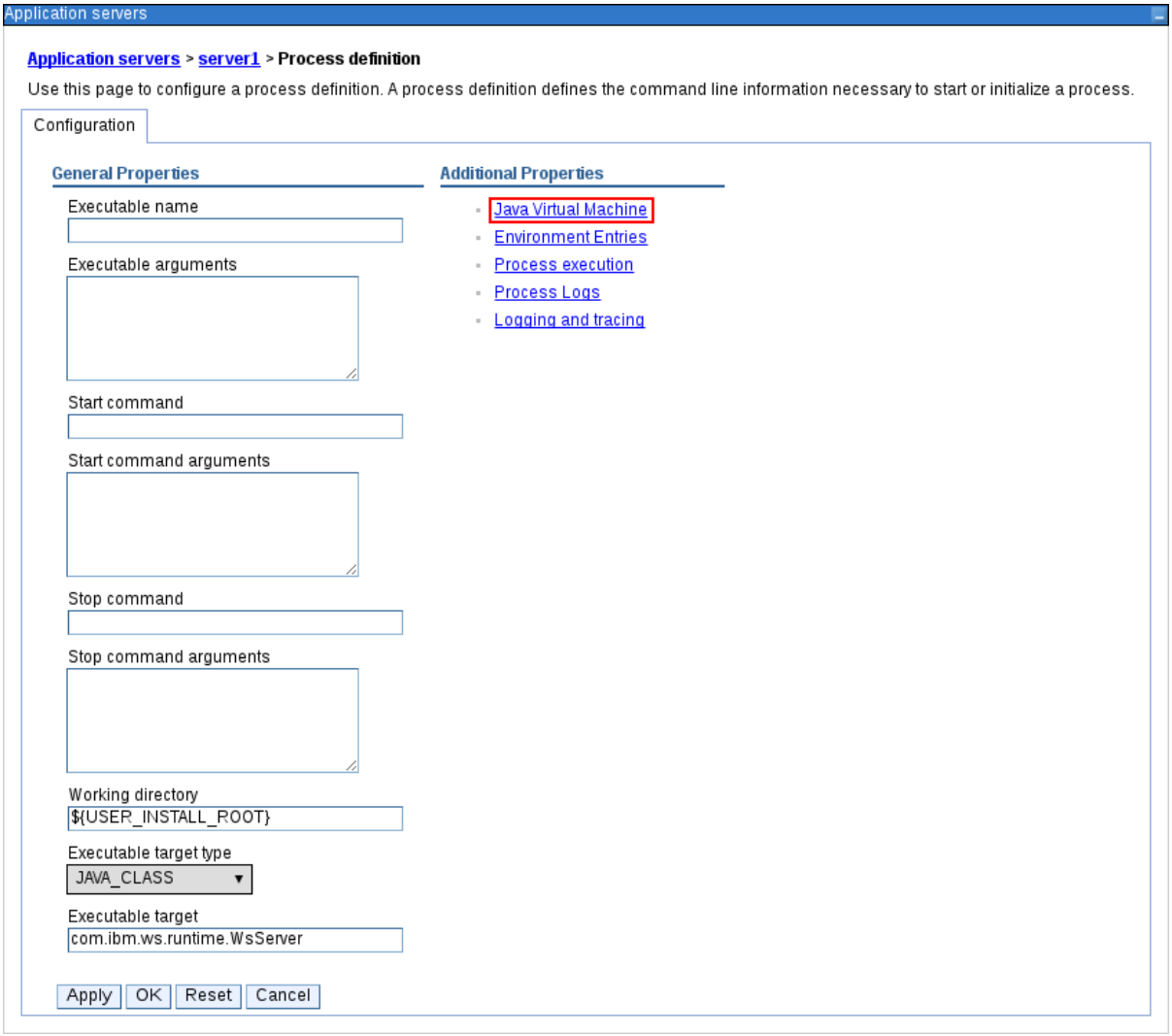
1. In the WebSphere Integrated Solutions Console, navigate to **Servers → Server Types → IBM WebSphere Application Servers**.
2. In the list of application servers, choose the server on which you are going to deploy KIE Server.
3. Under the **Server Infrastructure**, click **Java and Process Management → Process Definition**.

Figure 45.2. WebSphere configuration page



4. Under **Additional Properties**, click **Java Virtual Machine**

Figure 45.3. Process definition configuration page



This opens the configuration properties for the JVM that is used to start IBM WebSphere.

5. Set both the **Initial heap size** and **Maximum heap size** to **2048** and click **Apply** to increase the Java Virtual Machine (JVM) memory size. KIE Server has been tested with these values. If you do not increase the JVM memory size, IBM WebSphere Application Server freezes or causes deployment errors when deploying KIE Server.

6. Under **Additional Properties**, click **Custom Properties**.

7. Click **New → Custom JVM Properties** and add the following properties to IBM WebSphere:

Table 45.4. System properties for KIE Server

Name	Value	Description
kie.server.jms.queues.response	**jms/KIE.SERVER.RESPONSE**	The JNDI name of JMS queue for responses used by KIE Server.

Name	Value	Description
org.kie.server.domain	**WSLogin**	JAAS **LoginContext** domain used to authenticate users when using JMS.
org.jbpm.server.ext.disabled	**true**	Disables Business Central features, which are not supported in RHDM. If not set, KIE Server will work, but will show error messages during start up.
org.jbpm.ui.server.ext.disabled	**true**	Disables Business Central features, which are not supported in RHDM. If not set, KIE Server will work, but will show error messages during start up.
org.jbpm.case.server.ext.disabled	**true**	Disables Business Central features, which are not supported in RHDM. If not set, KIE Server will work, but will show error messages during start up.
org.jboss.logging.provider	**jdk**	This property is only required where a **CA SiteMinder TAI (SMTAI)** is installed in the environment. Using this property forces Hibernate to use **JDK** instead of **log4j** for logging within Dashbuilder. **CA SiteMinder TAI (SMTAI)** contains an old version of **log4j**, which causes conflicts.

8. Click **Save** to save the changes to the primary configuration.

45.4. STOPPING AND RESTARTING IBM WEBSPHERE APPLICATION SERVER

After you have configured all required system properties in IBM WebSphere Application Server, stop and restart the IBM server to ensure that the configurations are applied.

Procedure

In your command terminal, navigate to the IBM WebSphere Application Server /**bin** directory location that you specified during installation, and run the following commands to stop and restart IBM WebSphere to apply the configuration changes:

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

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

Replace **<SERVER_NAME>** with the IBM WebSphere Application Server name defined in **Servers → Server Types → IBM WebSphere Application Servers**of the WebSphere Integrated Solutions Console.

CHAPTER 46. INSTALLING AND RUNNING KIE SERVER WITH IBM WEBSPHERE APPLICATION SERVER

After you have configured all required system properties in IBM WebSphere Application Server, you can install KIE Server with IBM WebSphere to streamline Red Hat Decision Manager application management.

Prerequisites

- An IBM WebSphere Application Server instance is configured as described in Chapter 45, *Configuring IBM WebSphere Application Server for KIE Server* .

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:** 7.13.5
2. Download **Red Hat Process Automation Manager 7.13.5 KIE Server for All Supported EE7 Containers**.
3. Extract the **rhpam-7.13.5-kie-server-ee7.zip** archive to a temporary directory. In the following examples this directory is called ***TEMP_DIR***.
4. Repackage the **kie-server.war** directory:
   a. Navigate to the ***TEMP_DIR*/rhpam-7.13.5-kie-server-ee7/kie-server.war** directory.
   b. Select the contents of the ***TEMP_DIR*/rhpam-7.13.5-kie-server-ee7/kie-server.war** directory and create the **kie-server.zip** file.
   c. Rename **kie-server.zip** to **kie-server.war**. This is the file that you will use to deploy KIE Server.
   d. Optional: Copy the new **kie-server.war** file to a location that is more convenient to deploy from.
5. In the WebSphere Integrated Solutions Console, navigate to **Applications → Application Types → WebSphere Enterprise Applications**.
6. Click **InstCll**.
7. Navigate to the **kie-server.war** file that you repackaged and select it to upload.
8. Select **Fast Path** and click **Next**.
   The **Install New Application** wizard opens.
9. Change the **Application Name** to **kie-server** and click **Next**.
10. Map the KIE Server modules to servers according to your specific requirements and click **Next**.
11. For **Bind Listeners for Message-Driven Beans**, select **Activation Specification** for both beans, enter **jms/activation/KIE.SERVER.REQUEST** in the **Target Resource JNDI Name**

field, and enter the **jms/cf/KIE.SERVER.REQUEST** JNDI name for the **KIE.SERVER.REQUEST** connection factory.

12. In the **Map Virtual Hosts for Web Modules** section, keep the default values and click **Next**.
13. Set the context root to **kie-server**.
14. In the **Metadata for Modules** section, keep the default values and click **Next**.
15. Click **Finish** to install KIE Server and click **Save** to save the changes to the primary configuration.

46.1. CREATING THE KIE SERVER GROUP AND ROLE

After KIE Server is installed, you must create the **kie-server** group and a user.

Prerequisites

- KIE Server is installed on the IBM WebSphere Application Server instance.

Procedure

1. In the WebSphere Integrated Solutions Console, click **Users and Groups → Manage Groups**.
2. In the **Manage Groups** screen, click **Create**.
3. In the **Create a Group** screen, enter **kie-server** in the **Group name** box, then click **Create**.
4. To create a user to add to the **kie-server** group, click **Users and Groups → Manage Users**.
5. In the **Create a User** screen, complete the required information.
6. Click **Group Membership**.
7. In the **Group Membership** screen, click **kie-server**, move it to **Mapped To**, and click **Close**.
8. On the **Create a User** screen click **Create**.

46.2. MAPPING THE KIE SERVER GROUP AND ROLE

After KIE Server is installed, you must map the **kie-server** role to the **kie-server** group in the WebSphere Integrated Solutions Console to run KIE Server.

Prerequisites

- KIE Server is installed on the IBM WebSphere Application Server instance.
- IBM WebSphere Application Server has the **kie-server** group with at least one user.

Procedure

1. In the WebSphere Integrated Solutions Console, navigate to **Applications → Application Types → WebSphere Enterprise Applications** and select the newly installed **kie-server** application.
2. Under **Detail Properties**, click **Security Role to User/Group Mapping**

3. Select the **kie-server** role and click **Map Groups** to search for the **kie-server** group.
4. Move the **kie-server** group from the **Available** list to the **Selected** list and click **OK**.
   This mapping gives users in the IBM WebSphere Application Server **kie-server** group access to KIE Server.
5. Click **Save** to complete the mapping.

46.3. CONFIGURING CLASS LOADING FOR KIE SERVER

After KIE Server is installed, you must configure class loading to set parent classes to load last.

Procedure

1. Navigate to **Applications → Application Types → WebSphere Enterprise Applications** and click **kie-server**.
2. Click **Class Loading and Update Detection** under the **Detail Properties** heading on the left.
3. In the properties, change *Class Loader Order* to **Classes loaded with local class loader first (parent last)** and *WAR Class Loader Policy* to **Single class loader for application**
4. Save the changes to the primary configuration.

46.4. VERIFYING THE INSTALLATION

After you install KIE Server and define the KIE Server group mapping, verify that the server is running.

Prerequisites

- KIE Server is installed on the IBM WebSphere Application Server instance.
- You have set all required system properties for the headless Process Automation Manager controller.
- You have defined the KIE Server group mapping in IBM WebSphere Application Server.

Procedure

1. To verify that the server is running, complete one of the following tasks:
   - Navigate to the KIE Server URL **http://<HOST>:<PORT>/kie-server**.
   - Send a **GET** request to **http://<HOST>:<PORT>/kie-server/services/rest/server** to check whether the KIE Server REST API responds.
     In these examples, replace the following placeholders:
     - **<HOST>** is the ID or name of the headless Process Automation Manager controller, for example, **localhost** or **192.7.8.9**.
     - **<PORT>** is the port number of the KIE Server host, for example, **9060**.
2. If KIE Server is not running, stop and restart the IBM WebSphere Application Server instance and try again to access the KIE Server URL or API.

CHAPTER 47. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER WITH IBM WEBSPHERE APPLICATION SERVER

To use the KIE Server REST API or Java Client API to interact with KIE Server, install the headless Process Automation Manager controller with IBM WebSphere Application Server. The headless Process Automation Manager controller manages KIE Server configuration in a centralized way so that you can use the headless Process Automation Manager controller to create and maintain containers and perform other server-level tasks.

NOTE

For optimal results, install KIE Server and the headless Process Automation Manager controller on different servers in production environments. In development environments, you can install KIE Server and the headless Process Automation Manager controller on the same server.

Prerequisites

- The IBM WebSphere Application Server instance is configured as described in Chapter 45, *Configuring IBM WebSphere Application Server for KIE Server* .
- KIE Server is installed on the IBM WebSphere Application Server instance.
- You have sufficient user permissions to complete the installation.

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:** 7.13.5
2. Download **Red Hat Process Automation Manager 7.13.5 Add-Ons**
3. Extract the downloaded **rhpam-7.13.5-add-ons.zip** file to a temporary directory.
4. Extract the **rhpam-7.13.5-add-ons.zip** file to a temporary directory. In the following examples this directory is called ***TEMP_DIR***.
5. Extract the **rhpam-7.13.5-add-ons/rhpam-7.13.5-controller-ee7.zip** file.
6. Repackage the **controller.war** directory:
   a. Navigate to the ***TEMP_DIR*/rhpam-7.13.5-add-ons/rhpam-7.13.5-controller-ee7/controller.war** directory.
   b. Select the contents of the ***TEMP_DIR*/rhpam-7.13.5-add-ons/rhpam-7.13.5-controller-ee7/controller.war** directory and create the **controller.zip** file.
   c. Rename **controller.zip** to **controller.war**. This is the file that you will use to deploy the headless Process Automation Manager controller.

d. Optional: Copy the new **controller.war** file to a location that is more convenient to deploy from.

7. In the WebSphere Integrated Solutions Console, navigate to **Applications → Application Types → WebSphere Enterprise Applications**.
8. Click **Install**.
9. Navigate to the **controller.war** file that you repackaged and select it to upload.
10. Select **Fast Path** and click **Next**.
The **Install New Application** wizard opens.
11. Change the **Application Name** to **controller** and click **Next**.
12. Map the headless Process Automation Manager controller modules to servers according to your specific requirements and click **Next**.
13. For **Bind Listeners for Message-Driven Beans**, select **Activation Specification** for both beans, enter **jms/activation/KIE.SERVER.REQUEST** in the **Target Resource JNDI Name** field, and enter the **jms/cf/KIE.SERVER.REQUEST** JNDI name for the **KIE.SERVER.REQUEST** connection factory.
14. In the **Map Virtual Hosts for Web Modules** section, keep the default values and click **Next**.
15. Set the context root to **controller**.
16. In the **Metadata for Modules** section, keep the default values and click **Next**.
17. Click **Finish** to install the headless Process Automation Manager controller and click **Save** to save the changes to the primary configuration.

47.1. CONFIGURING CLASS LOADING FOR THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

After the headless Process Automation Manager controller is installed, you must configure KIE Server class loading to set parent classes to load last.

Procedure

1. Navigate to **Applications → Application Types → WebSphere Enterprise Applications** and click **kie-server**.
2. Click **Class Loading and Update Detection** under the **Detail Properties** heading on the left.
3. In the properties, change *Class Loader Order* to **Classes loaded with local class loader first (parent last)** and *WAR Class Loader Policy* to **Single class loader for application**
4. Save the changes to the primary configuration.

47.2. SETTING SYSTEM PROPERTIES FOR THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

After you install the headless Process Automation Manager controller, set the system properties listed in this section on your application server or servers to enable proper interaction with the headless Process Automation Manager controller.

NOTE

For optimal results, install KIE Server and the headless Process Automation Manager controller on different servers in production environments. In development environments, you can install KIE Server and the headless Process Automation Manager controller on the same server. In either case, be sure to make these property changes on all application servers where the headless Process Automation Manager controller is installed.

Prerequisites

- KIE Server and the headless Process Automation Manager controller are installed on the application server instance.

Procedure

1. Specify the following JVM property values on the application server instance where the headless Process Automation Manager controller is installed:

Table 47.1. Required properties for the headless Process Automation Manager controller

Name	Requirement
org.kie.server.user	A user with the **kie-server** role
org.kie.server.pwd	The password for the user specified in the **org.kie.server.user** property

2. Specify the following JVM property values on the application server instance where KIE Server is installed:

Table 47.2. Required properties for KIE Server when headless Process Automation Manager controller is installed

Name	Requirement
org.kie.server.controller.user	A user with the **kie-server** role
org.kie.server.controller.pwd	The password for the user specified for the **org.kie.server.controller.user** property
org.kie.server.id	The ID or name of the KIE Server installation, such as **rhdm700-decision-server-1**
org.kie.server.location	The URL of KIE Server, **http://<HOST>:<PORT>/kie-server/services/rest/server**

Name	Requirement
org.kie.server.controller	The URL of the headless Process Automation Manager controller, **http://<HOST>:<PORT>/controller/rest/controller**

<HOST> is the ID or name of the KIE Server host, for example, **localhost** or **192.7.8.9**.

<PORT> is the port number of the KIE Server host, for example, **7001**.

47.3. MAPPING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER GROUP AND ROLE

After the headless Process Automation Manager controller is installed, in the WebSphere Integrated Solutions Console, you must create the **kie-server** group and then map the **kie-server** role to the **kie-server** group.

Prerequisites

- The headless Process Automation Manager controller is installed on the IBM WebSphere Application Server instance.

Procedure

1. In the WebSphere Integrated Solutions Console, create the **kie-server** group and a user for that group, as described in *Creating the KIE Server group and role* .
2. Navigate to **Applications → Application Types → WebSphere Enterprise Applications** and select the newly installed headless Process Automation Manager controller.
3. Under **Detail Properties**, click **Security Role to User/Group Mapping**
4. Select the **kie-server** role and click **Map Groups** to search for the **kie-server** group.
5. Move the **kie-server** group from the **Available** list to the **Selected** list and click **OK**.
   This mapping gives the previously created administrator user access to Decision Manager.
6. Click **Save** to complete the mapping.

47.4. VERIFYING THE INSTALLATION

After you install the headless Process Automation Manager controller and define the required system properties and role requirements on the application server, verify that the headless Process Automation Manager controller works correctly.

Prerequisites

- KIE Server and the headless Process Automation Manager controller are installed on the application server instance.
- You have set all required system properties and role requirements for the headless Process Automation Manager controller on the application server.

Procedure

In your command terminal, enter the following command to verify that the headless Process Automation Manager controller is working:

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

<HOST> is the ID or name of the headless Process Automation Manager controller host, for example, **localhost** or **192.7.8.9**.

<PORT> is the port number of the headless Process Automation Manager controller host, for example, **7001**.

<CONTROLLER> and **<CONTROLLER_PWD>** are the user credentials that you created in this section.

The command should return information about the KIE Server instance.

NOTE

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

If the headless Process Automation Manager controller is not running, stop and restart the application server instance and try again to access the headless Process Automation Manager controller URL or API.

CHAPTER 48. CONFIGURING AN EMBEDDED DECISION ENGINE IN IBM WEBSPHERE APPLICATION SERVER

A decision engine is a light-weight rule engine that enables you to execute your decisions and business processes. A decision engine can be part of a Red Hat Decision Manager application or it can be deployed as a service through OpenShift, Kubernetes, and Docker. You can embed a decision engine in a Red Hat Decision Manager application through the API or as a set of contexts and dependency injection (CDI) services.

If you intend to use an embedded engine with your Red Hat Process Automation Manager application, you must add Maven dependencies to your project by adding 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. 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>7.13.5.redhat-00002</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.

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

     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 KIE Server, declare the following dependencies:

  Client application KIE 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>
```

3. If you use a decision engine with persistence support in your project, you must declare the following hibernate dependencies in the **dependencyManagement** section of your **pom.xml** file by copying the **version.org.hibernate-4ee7** property from the Red Hat Business Automation BOM file:

Hibernate dependencies in decision engine with persistence

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

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

CHAPTER 49. SECURING PASSWORDS WITH A KEYSTORE

You can use a keystore to encrypt passwords that are used for communication between Business Central and KIE Server. You should encrypt both controller and KIE Server passwords. If Business Central and KIE 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.

NOTE

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

Prerequisites

- KIE Server is installed in IBM WebSphere Application Server.
- A KIE Server user with the **kie-server** role has been created, as described in Section 46.1, “Creating the KIE Server group and role”.
- Java 8 or higher is installed.

Procedure

1. Create a JCEKS keystore.
2. When prompted, enter the password for the KIE Server user that you created.
3. Set the system properties listed in the following table:

Table 49.1. System properties used to load a KIE 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_ALIAS>**	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_ALIAS>**	Alias of the key for default REST Process Automation Controller where the password is stored

System property	Placeholder	Description
kie.keystore.key.ctrl.pwd	**<KEY_CONTROL_PWD>**	Password of the alias for default REST Process Automation Controller with the stored password

4. Start KIE Server to verify the configuration.

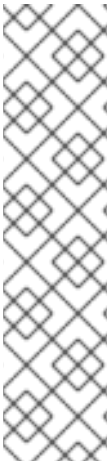
CHAPTER 50. NEXT STEPS

- *Getting started with decision services*
- *Designing a decision service using guided decision tables*

PART VI. INSTALLING AND CONFIGURING KIE SERVER ON ORACLE WEBLOGIC SERVER

Red Hat Decision Manager is a subset of Red Hat Process Automation Manager. Starting with this release, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files. There are no Decision Manager artifacts. The Red Hat Decision Manager subscription, support entitlements, and fees remain the same. Red Hat Decision Manager subscribers will continue to receive full support for the decision management and optimization capabilities of Red Hat Decision Manager. The business process management (BPM) capabilities of Red Hat Process Automation Manager are exclusive to Red Hat Process Automation Manager subscribers. They are available for use by Red Hat Decision Manager subscribers but with development support services only. Red Hat Decision Manager subscribers can upgrade to a full Red Hat Process Automation Manager subscription at any time to receive full support for BPM features.

As a system administrator, you can configure your Oracle WebLogic Server for Red Hat KIE Server and install KIE Server on that Oracle server instance.



NOTE

Support for Red Hat Decision Manager on Oracle WebLogic Server is now in the maintenance phase. Red Hat will continue to support Red Hat Decision Manager on Oracle WebLogic Server with the following limitations:

- Red Hat will not release new certifications or software functionality.
- Red Hat will release only qualified security patches that have a critical impact and mission-critical bug fix patches.

In the future, Red Hat might direct customers to migrate to new platforms and product components that are compatible with the Red Hat hybrid cloud strategy.

Prerequisites

- An Oracle WebLogic Server instance version 12.2.1.3.0 or later is installed. For complete installation instructions, see the Oracle WebLogic Server product page .
- You have access to the Oracle WebLogic Server Administration Console, usually at **http://<HOST>:7001/console**.

CHAPTER 51. KIE SERVER

KIE Server is the server where the rules and other artifacts for Red Hat Process Automation Manager are stored and run. KIE Server is a standalone built-in component that can be used to instantiate and execute rules through interfaces available for REST, Java Message Service (JMS), or Java client-side applications, and Red Hat build of OptaPlanner functionality through solvers.

Created as a web deployable WAR file, KIE Server can be deployed on any web container. The current version of KIE Server is included with default extensions for both Red Hat Decision Manager and Red Hat Process Automation Manager.

KIE Server has a low footprint with minimal memory consumption and therefore can be deployed easily on a cloud instance. Each instance of this server can open and instantiate multiple containers, which enables you to execute multiple rule services in parallel.

KIE Server can be integrated with other application servers, such as Oracle WebLogic Server or IBM WebSphere Application Server, to streamline Red Hat Process Automation Manager application management.

CHAPTER 52. ORACLE WEBLOGIC SERVER

Oracle WebLogic Server is a Java EE application server that provides a standard set of APIs for creating distributed Java applications that can access a wide variety of services, such as databases, messaging services, and connections to external enterprise systems. User clients access these applications using web browser clients or Java clients.

CHAPTER 53. INSTALLING AND RUNNING ORACLE WEBLOGIC SERVER

Oracle WebLogic Server must be installed and running for you to apply many of the configurations that accommodate KIE Server. This section describes how to install and start Oracle WebLogic Server in a standalone Oracle WebLogic Server domain.

For the most up-to-date and detailed installation instructions, see the Oracle WebLogic Server product page.

NOTE

If you are already running an instance of Oracle WebLogic Server that uses the same listener port as the one to be used by the server you are starting, you must stop the first server before starting the second server.

Procedure

1. Download Oracle WebLogic Server 12.2.1.3.0 or later from the Oracle WebLogic Server Downloads page.
2. Sign in to the target system and verify that a certified JDK already exists on your system. The installer requires a certified JDK. For system requirements, see Oracle Fusion Middleware Systems Requirements and Specifications. To download the JDK, see the "About JDK Requirements for an Oracle Fusion Middleware Installation" section in *Planning an Installation of Oracle Fusion Middleware*.
3. Navigate to the directory where you downloaded the installation program.
4. To launch the installation program, run **java -jar** from the JDK directory on your system, as shown in the following examples:
   On UNIX-based operating systems, enter the following command:

   ```
   /home/Oracle/jdk/jdk1.8.0_131/bin/java -jar fmw_12.2.1.3.0_wls_generic.jar
   ```

   On Windows operating systems, enter the following command:

   ```
   C:\Program Files\Java\jdk1.8.0_131\bin\java -jar fmw_12.2.1.3.0_wls_generic.jar
   ```

   Replace the JDK location in these examples with the actual JDK location on your system.
5. Follow the installation wizard prompts to complete the installation.
6. After the installation is complete, navigate to the ***WLS_HOME*/user_projects/<DOMAIN_NAME>** directory where **<DOMAIN_NAME>** is the domain directory. In the following example, **mydomain** is the domain directory:

   ```
   WLS\user_projects\mydomain
   ```

7. Enter one of the following commands to start Oracle WebLogic Server:
   On UNIX-based operating systems, enter the following command:

   ```
   startWebLogic.sh
   ```

On Windows operating systems, enter the following command:

```
startWebLogic.cmd
```

The startup script displays a series of messages, and finally displays a message similar to the following:

```
<Dec 8, 2017 3:50:42 PM PDT> <Notice> <WebLogicServer> <000360> <Server started in RUNNING mode>
```

8. Open the following URL in a web browser:

```
http://<HOST>:<PORT>/console
```

In this command, replace the following place holders:

- Replace **<HOST>** with the system name or IP address of the host server.
- Replace **<PORT>** with the number of the port on which the host server is listening for requests (7001 by default).

For example, to start the Administration Console for a local instance of Oracle WebLogic Server running on your system, enter the following URL in a web browser:

```
http://localhost:7001/console/
```

If you started the Administration Console using secure socket layer (SSL), you must add **s** after **http**, as follows: **https://<HOST>:<PORT>/console**

9. When the login page of the WebLogic Administration Console appears, enter your administrative credentials.

CHAPTER 54. CONFIGURING ORACLE WEBLOGIC SERVER FOR KIE SERVER

Before you deploy KIE Server with Oracle WebLogic Server, you must configure system properties, security settings, JMS requirements, and other properties on Oracle WebLogic Server. These configurations promote an optimal integration with KIE Server.

Prerequisites

- Oracle WebLogic Server is installed and running.
- You are logged in to the WebLogic Administration Console.

54.1. CONFIGURING THE KIE SERVER GROUP AND USERS

You must assign users to a **kie-server** group in the WebLogic Administration Console to enable the container-managed authentication mechanisms in Oracle WebLogic Server.

Procedure

1. In the WebLogic Administration Console, click **Security Realms**.
2. Choose your desired security realm or click **New** to create a new security realm.
3. Navigate to **Users and Groups** → **Groups** → **New** and create the **kie-server** group.
4. Navigate to **Users** → **New** and create a new user.
5. Enter a user, such as **server-user**, and a password for this new user and click **OK**.

IMPORTANT

Make sure that the selected user name does not conflict with any known title of a role or a group. For example, if there is a role called **kie-server**, then do not create a user with the user name **kie-server**.

6. Click the newly created user, then return to the **Groups** tab.
7. Use the selection tool to move the **kie-server** group from the **Available** field to the **Chosen** field, and click **Save**.

54.2. CONFIGURING JAVA MESSAGE SERVICE (JMS)

The Java Message Service (JMS) is a Java API that KIE Server uses to exchange messages with other application servers such as Oracle WebLogic Server and IBM WebSphere Application Server. You must configure your application server to send and receive JMS messages through KIE Server to ensure collaboration between the two servers.

54.2.1. Create a JMS server

Create a JMS server to use JMS with KIE Server and Oracle WebLogic Server.

Procedure

1. In the WebLogic Administration Console, navigate to **Services → Messaging → JMS Servers**.
2. Click **New** to create a new JMS server.
3. Enter a name for your JMS server and click **Next**.
4. Select the target server chosen for the KIE Server deployment.
5. Click **Finish**.

54.2.2. Create a JMS module

You must create a JMS module to store your JMS resources, such as connection factories and queues.

Prerequisites

- You have created a JMS server.

Procedure

1. In the WebLogic Administration Console, navigate to **Services → Messaging → JMS Modules**.
2. Click **New** to create a module.
3. Enter a module name and click **Next**.
4. Select the target server chosen for the KIE Server deployment and click **Finish**.
5. Click the newly created module name and then click **Subdeployments**.
6. Click **New** to create a subdeployment for your module.
7. Give your subdeployment a name and click **Next**.
8. Select the check box to choose the previously created JMS server.
9. Click **Finish** to complete the subdeployment configuration.

54.2.3. Create JMS connection factories

To enable messaging with KIE Server, you must create certain JMS connection factories for sending and receiving messages.

Prerequisites

- You have created a JMS server.
- You have created a JMS module.

Procedure

1. In the WebLogic Administration Console, navigate to **Services → Messaging → JMS Modules** to see a list of JMS modules.

2. Select your previously created module and click **New** to create a new JMS resource.
3. Select **Connection Factory** and click **Next**.
4. For each of rthe equired connection factories listed in the following table, enter the name of the connection factory (for example, **KIE.SERVER.REQUEST**) and the JNDI name (for example, **jms/cf/KIE.SERVER.REQUEST**) and click **Next**. The connection factory automatically selects the servers assigned to the JMS Module as the default.

   Table 54.1. Required JMS connection factories for KIE Server

Name	Default value	Used for
KIE.SERVER.REQUEST	**jms/cf/KIE.SERVER.REQUEST**	Sending all requests to KIE Server
KIE.SERVER.RESPONSE	**jms/cf/KIE.SERVER.RESPONSE**	Receiving all responses produced by KIE Server

5. Click **Finish** to add the connection factory, and repeat for each required factory.

54.2.4. Create JMS queues

JMS queues are the destination end points for point-to-point messaging. You must create certain JMS queues to enable JMS messaging with KIE Server.

Prerequisites

- You have created a JMS server.
- You have created a JMS module.

Procedure

1. In the WebLogic Administration Console, navigate to **Services → Messaging → JMS Modules** to see the list of JMS modules.
2. Select your previously created module, then click **New** to create a new JMS resource.
3. Select **Queue** and click **Next**.
4. For each of the required queues listed in the following table, enter the name of the queue (for example, **KIE.SERVER.REQUEST**) and the JNDI name (for example, **jms/KIE.SERVER.REQUEST**) and then click **Next**.

   Table 54.2. Required JMS queues for KIE Server

Name	Default value	Used for
KIE.SERVER.REQUEST	**jms/KIE.SERVER.REQUEST**	Sending all requests to KIE Server

Name	Default value	Used for
KIE.SERVER.RESPONSE	**jms/KIE.SERVER.RESPONSE**	Receiving all responses produced by KIE Server

5. Choose the JMS module subdeployment that connects to the JMS server.
6. Click **Finish** to add the queue, and repeat for each required queue.

54.3. SETTING SYSTEM PROPERTIES IN ORACLE WEBLOGIC SERVER

Set the system properties listed in this section on your Oracle WebLogic Server before you deploy KIE Server.

Procedure

1. Set the following system property to increase the Java Virtual Machine (JVM) memory size:

```
USER_MEM_ARGS=-Xms512m -Xmx1024m
```

   If you do not increase the JVM memory size, Oracle WebLogic Server freezes or causes deployment errors when deploying KIE Server.

2. Specify the following system properties for KIE Server on the Oracle WebLogic Server instance:

Table 54.3. System properties for KIE Server

Name	Value	Description
kie.server.jms.queues.response	**jms/KIE.SERVER.RESPONSE**	The JNDI name of JMS queue for responses used by KIE Server.
org.kie.server.domain	**OracleDefaultLoginConfiguration**	JAAS **LoginContext** domain used to authenticate users when using JMS.
org.jbpm.server.ext.disabled	**true**	Disables Business Central features, which are not supported in RHDM. If not set, KIE Server will work, but will show error messages during start up.
org.jbpm.ui.server.ext.disabled	**true**	Disables Business Central features, which are not supported in RHDM. If not set, KIE Server will work, but will show error messages during start up.

Name	Value	Description
org.jbpm.case.server.ext.disabled	**true**	Disables Business Central features, which are not supported in RHDM. If not set, KIE Server will work, but will show error messages during start up.

3. Set the same property values in the **JAVA_OPTIONS** environment variable:

```
JAVA_OPTIONS="-Dkie.server.jms.queues.response=jms/KIE.SERVER.RESPONSE
 -Dorg.kie.server.domain=OracleDefaultLoginConfiguration
 -Dorg.jbpm.server.ext.disabled=true
 -Dorg.jbpm.ui.server.ext.disabled=true
 -Dorg.jbpm.case.server.ext.disabled=true"
```

54.4. STOPPING AND RESTARTING ORACLE WEBLOGIC SERVER

After you have configured all required system properties in Oracle WebLogic Server, stop and restart the Oracle server to ensure that the configurations are applied.

Procedure

1. In the WebLogic Administration Console, navigate to **Change Center → Lock & Edit**
2. Under **Domain Structure**, click **Environment → Servers → Control**.
3. Select the server that you want to stop and click **Shutdown**.
4. Select **When Work Completes** to gracefully shut down the server or select **Force Shutdown Now** to stop the server immediately without completing ongoing tasks.
5. On the **Server Life Cycle Assistant** pane, click **Yes** to complete the shutdown.
6. After the shutdown is complete, navigate to the domain directory in the command terminal, ***WLS_HOME*/user_projects/<DOMAIN_NAME>**. For example:

```
WLS\user_projects\mydomain
```

7. Enter one of the following commands to restart Oracle WebLogic Server to apply the new configurations:
   On UNIX-based operating systems:

```
startWebLogic.sh
```

   On Windows operating systems:

```
startWebLogic.cmd
```

8. Open the Administration Console in a web browser (for example, **http://localhost:7001/console/**) and log in with your credentials.

CHAPTER 55. INSTALLING KIE SERVER WITH ORACLE WEBLOGIC SERVER

After you have configured all required system properties in Oracle WebLogic Server, you can install KIE Server with Oracle WebLogic Server to streamline Red Hat Decision Manager application management.

Prerequisites

- An Oracle WebLogic Server instance is configured as described in Chapter 54, *Configuring Oracle WebLogic Server for KIE Server*.

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:** 7.13.5
2. Download **Red Hat Process Automation Manager 7.13.5 KIE Server for All Supported EE7 Containers**.
3. Extract the downloaded **rhpam-7.13.5-kie-server-ee7.zip** file to a temporary directory.
4. In the WebLogic Administration Console, navigate to **Deployments** to view all existing applications.
5. Click **Install**.
6. Navigate to the temporary directory where you downloaded and extracted the **rhpam-7.13.5-kie-server-ee7.zip** file, and go to **rhpam-7.13.5-kie-server-ee7/kie-server.war**.
7. Select the **kie-server.war** file and click **Next** to continue.
8. Select **Install this deployment as an application**as the targeting style and click **Next**.
9. Set the application name to **kie-server** and set the security model to **DD Only**. Leave the remaining options as default and click **Next** to continue.
10. In the **Additional Configuration** section, choose **No, I will review the configuration later**and click **Finish**.

55.1. VERIFYING THE KIE SERVER INSTALLATION ON ORACLE WEBLOGIC SERVER

After you have installed KIE Server on Oracle WebLogic Server, verify that the installation was successful.

Prerequisites

- An Oracle WebLogic Server instance is configured as described in Chapter 54, *Configuring Oracle WebLogic Server for KIE Server*.

- KIE Server is installed as described in Chapter 55, *Installing KIE Server with Oracle WebLogic Server*.

Procedure

1. Enter the KIE Server URL **http://<HOST>:<PORT>/kie-server/services/rest/server** in a web browser.
2. Verify that KIE Server is running.
   If KIE Server is not running, stop and restart the Oracle WebLogic Server instance and try again to access the KIE Server URL.

CHAPTER 56. INSTALLING AND RUNNING THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER WITH ORACLE WEBLOGIC SERVER

To use the KIE Server REST API or Java Client API to interact with KIE Server, install the headless Process Automation Manager controller with Oracle WebLogic Server. The headless Process Automation Manager controller manages KIE Server configuration in a centralized way so that you can use the headless Process Automation Manager controller to create and maintain containers and perform other server-level tasks.

Prerequisites

- The Oracle WebLogic Server instance is configured as described in Chapter 54, *Configuring Oracle WebLogic Server for KIE Server*.
- KIE Server is installed on the Oracle WebLogic Server instance.
- You have sufficient user permissions to complete the installation.

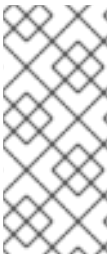
Procedure

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:** 7.13.5
2. Download **Red Hat Process Automation Manager 7.13.5 Add-Ons**.
3. Extract the downloaded **rhpam-7.13.5-add-ons.zip** file to a temporary directory.
4. In the WebLogic Administration Console, navigate to **Security Realms → Users and Groups**.
5. In the **kie-server** group that you created previously, create a user for the headless Process Automation Manager controller, such as **controller**, and a password for this new user and click **OK**. For more information about creating groups and users, see Section 54.1, "Configuring the KIE Server group and users".
6. Navigate to **Deployments** to view all existing applications.
7. Click **Install**.
8. Navigate to the temporary directory where you downloaded and extracted the **rhpam-7.13.5-add-ons.zip** file, and go to **rhpam-7.13.5-add-ons/rhpam-7.13.5-controller-ee7.zip/controller.war**.
9. Select the **controller.war** file and click **Next** to continue.
10. Select **Install this deployment as an application** as the targeting style and click **Next**.
11. Keep the application name as **controller** and set the security model to **DD Only**. Leave the remaining options as default and click **Next** to continue.

12. In the **Additional Configuration** section, choose **No, I will review the configuration later** and click **Finish**.

56.1. SETTING SYSTEM PROPERTIES FOR THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

After you install the headless Process Automation Manager controller, set the system properties listed in this section on your application server or servers to enable proper interaction with the headless Process Automation Manager controller.



NOTE

For optimal results, install KIE Server and the headless Process Automation Manager controller on different servers in production environments. In development environments, you can install KIE Server and the headless Process Automation Manager controller on the same server. In either case, be sure to make these property changes on all application servers where the headless Process Automation Manager controller is installed.

Prerequisites

- KIE Server and the headless Process Automation Manager controller are installed on the application server instance.

Procedure

1. Specify the following JVM property values on the application server instance where the headless Process Automation Manager controller is installed:

Table 56.1. Required properties for the headless Process Automation Manager controller

Name	Requirement
org.kie.server.user	A user with the **kie-server** role
org.kie.server.pwd	The password for the user specified in the **org.kie.server.user** property

2. Specify the following JVM property values on the application server instance where KIE Server is installed:

Table 56.2. Required properties for KIE Server when headless Process Automation Manager controller is installed

Name	Requirement
org.kie.server.controller.user	A user with the **kie-server** role
org.kie.server.controller.pwd	The password for the user specified for the **org.kie.server.controller.user** property

Name	Requirement
org.kie.server.id	The ID or name of the KIE Server installation, such as **rhdm700-decision-server-1**
org.kie.server.location	The URL of KIE Server, **http://<HOST>:<PORT>/kie-server/services/rest/server**
org.kie.server.controller	The URL of the headless Process Automation Manager controller, **http://<HOST>:<PORT>/controller/rest/controller**

<HOST> is the ID or name of the KIE Server host, for example, **localhost** or **192.7.8.9**.

<PORT> is the port number of the KIE Server host, for example, **7001**.

56.2. VERIFYING THE INSTALLATION

After you install the headless Process Automation Manager controller and define the required system properties and role requirements on the application server, verify that the headless Process Automation Manager controller works correctly.

Prerequisites

- KIE Server and the headless Process Automation Manager controller are installed on the application server instance.
- You have set all required system properties and role requirements for the headless Process Automation Manager controller on the application server.

Procedure

In your command terminal, enter the following command to verify that the headless Process Automation Manager controller is working:

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

<HOST> is the ID or name of the headless Process Automation Manager controller host, for example, **localhost** or **192.7.8.9**.

<PORT> is the port number of the headless Process Automation Manager controller host, for example, **7001**.

<CONTROLLER> and **<CONTROLLER_PWD>** are the user credentials that you created in this section.

The command should return information about the KIE Server instance.

NOTE

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

If the headless Process Automation Manager controller is not running, stop and restart the application server instance and try again to access the headless Process Automation Manager controller URL or API.

CHAPTER 57. CONFIGURING AN EMBEDDED DECISION ENGINE IN ORACLE WEBLOGIC SERVER

A decision engine is a light-weight rule engine that enables you to execute your decisions and business processes. A decision engine can be part of a Red Hat Decision Manager application or it can be deployed as a service through OpenShift, Kubernetes, and Docker. You can embed a decision engine in a Red Hat Decision Manager application through the API or as a set of contexts and dependency injection (CDI) services.

If you intend to use an embedded engine with your Red Hat Process Automation Manager application, you must add Maven dependencies to your project by adding 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. 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>7.13.5.redhat-00002</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.

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

     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 KIE Server, declare the following dependencies:

  Client application KIE 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>
```

3. If you use a decision engine with persistence support in your project, you must declare the following hibernate dependencies in the **dependencyManagement** section of your **pom.xml** file by copying the **version.org.hibernate-4ee7** property from the Red Hat Business Automation BOM file:

Hibernate dependencies in decision engine with persistence

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

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

CHAPTER 58. SECURING PASSWORDS WITH A KEYSTORE

You can use a keystore to encrypt passwords that are used for communication between Business Central and KIE Server. You should encrypt both controller and KIE Server passwords. If Business Central and KIE 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 KIE Server is not configured with JCEKS, KIE Server passwords are stored in system properties in plain text form.

Prerequisites

- KIE Server is installed in Oracle WebLogic Server.
- A KIE Server user with the **kie-server** role has been created, as described in Section 54.1, "Configuring the KIE Server group and users".
- Java 8 or higher is installed.

Procedure

1. 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

2. When prompted, enter the password for the KIE Server user that you created.
3. Set the system properties listed in the following table:

Table 58.1. System properties used to load a KIE 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**

System property	Placeholder	Description
kie.keystore.keyStorePwd	**<KEYSTORE_PWD>**	Password for the JCEKS
kie.keystore.key.server.alias	**<KEY_SERVER_ALIAS>**	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_ALIAS>**	Alias of the key for default REST Process Automation Controller where the password is stored
kie.keystore.key.ctrl.pwd	**<KEY_CONTROL_PWD>**	Password of the alias for default REST Process Automation Controller with the stored password

4. Start KIE Server to verify the configuration.

CHAPTER 59. NEXT STEPS

- *Getting started with decision services*
- *Designing a decision service using guided decision tables*

PART VII. PATCHING AND UPGRADING RED HAT DECISION MANAGER

Starting with release 7.13, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files.

You can apply updates to Red Hat Decision Manager release 7.12 and earlier or Red Hat Process Automation Manager 7.13 as they become available in the Red Hat Customer Portal to keep your distribution current with the latest enhancements and fixes. Red Hat provides update tools and product notifications for new product releases so you can more readily apply helpful updates to your installation environment.

Prerequisites

- You have a Red Hat Customer Portal account.
- Red Hat Decision Manager or Red Hat Process Automation Manager is installed. For installation options, see *Planning a Red Hat Decision Manager installation* .

NOTE

If you are using Red Hat Decision Manager 7.10 and you want to upgrade to Red Hat Process Automation Manager 7.13, see *Patching and upgrading Red Hat Decision Manager 7.10* .

CHAPTER 60. RED HAT DECISION MANAGER VERSIONING

Red Hat Process Automation Manager versions are designated with a numerical **Major.Minor.Patch** format, such as **7.5.1**. In this example, the major release is **7.x.x**, the minor release is **7.5.x**, and the patch release is **7.5.1**. Major releases often require data migration, while minor release upgrades and patch updates are typically managed with update tools provided with the Red Hat Decision Manager release artifacts.

NOTE

Starting with release 7.13, the distribution files for Red Hat Decision Manager are replaced with Red Hat Process Automation Manager files.

The following are the general types of releases for Red Hat Decision Manager:

Major release migrations

Major releases of Red Hat Decision Manager include substantial enhancements, security updates, bug fixes, and possibly redesigned features and functions. Data migration is typically required when an application is moved from one major release to another major release, such as from Red Hat JBoss BRMS 6.4.x to Red Hat Decision Manager 7.0. Automated migration tools are often provided with new major versions of Red Hat Decision Manager to facilitate migration, but some manual effort is likely required for certain data and configurations. The supported migration paths are specified in product announcements and documentation.

For example migration instructions, see *Migrating from Red Hat JBoss BRMS 6.4 to Red Hat Decision Manager 7.0*.

Minor release upgrades

Minor releases of Red Hat Decision Manager include enhancements, security updates, and bug fixes. Data migration may be required when an application is moved from one minor release to another minor release, such as from Red Hat Decision Manager 7.5.x to 7.6. Automated update tools are often provided with both patch updates and new minor versions of Red Hat Decision Manager to facilitate updating certain components of Red Hat Decision Manager, such as Business Central, KIE Server, and the headless Process Automation Manager controller. Other Red Hat Decision Manager artifacts, such as the decision engine and standalone Business Central, are released as new artifacts with each minor release and you must reinstall them to apply the update.

Before you upgrade to a new minor release, apply the latest patch update to your current version of Red Hat Decision Manager to ensure that the minor release upgrade is successful.

Patch updates

Patch updates of Red Hat Decision Manager include the latest security updates and bug fixes. Scheduled patch updates contain all previously released patch updates for that minor version of the product, so you do not need to apply each patch update incrementally in order to apply the latest update. For example, you can update Red Hat Decision Manager 7.5.0 or 7.5.1 to Red Hat Decision Manager 7.5.2. However, for optimal Red Hat Decision Manager performance, apply product updates as they become available. Occasionally, Red Hat might release unscheduled patch updates outside the normal update cycle of the existing product. These may include security or other updates provided by Red Hat Global Support Services (GSS) to fix specific issues, and may not be cumulative updates.

Automated update tools are often provided with both patch updates and new minor versions of Red Hat Decision Manager to facilitate updating certain components of Red Hat Decision Manager, such as Business Central, KIE Server, and the headless Process Automation Manager controller. Other

Red Hat Decision Manager artifacts, such as the decision engine and standalone Business Central, are released as new artifacts with each minor release and you must reinstall them to apply the update.

To ensure optimal transition between releases and to keep your Red Hat Decision Manager distribution current with the latest enhancements and fixes, apply new product releases and updates to Red Hat Decision Manager as they become available in the Red Hat Customer Portal. Consider also enabling product notifications in the Red Hat Customer Portal.

CHAPTER 61. ENABLING NOTIFICATIONS IN THE RED HAT CUSTOMER PORTAL

You can enable notifications in the Red Hat Customer Portal to receive product updates and announcements. These notifications inform you of updated or added documentation, product releases, and patch updates related to your installation. With notifications enabled, you can more readily apply product updates as they become available in the Red Hat Customer Portal to keep your distribution current with the latest enhancements and fixes.

Prerequisites

- You have a Red Hat Customer Portal account and are logged in.

Procedure

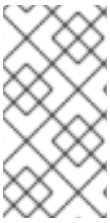
1. In the top-right corner of the Red Hat Customer Portal window, click your profile name and click **Notifications**.
2. Select the **Notifications** tab and click **Manage Notifications**.
3. Next to **Follow**, select **Products** from the drop-down menu, and then select **Red Hat Process Automation Manager** or **Red Hat Decision Manager** from the next drop-down menu that appears.
4. Click **Save Notification** to finish.

You can add notifications for any other products as needed in the same way.

CHAPTER 62. APPLYING PATCH UPDATES AND MINOR RELEASE UPGRADES TO RED HAT PROCESS AUTOMATION MANAGER

Automated update tools are often provided with both patch updates and new minor versions of Red Hat Process Automation Manager to facilitate updating certain components of Red Hat Process Automation Manager, such as Business Central, KIE Server, and the headless Process Automation Manager controller. Other Red Hat Process Automation Manager artifacts, such as the decision engine and standalone Business Central, are released as new artifacts with each minor release and you must reinstall them to apply the update.

You can use the same automated update tool to apply both patch updates and minor release upgrades to Red Hat Process Automation Manager 7.13. Patch updates of Red Hat Process Automation Manager, such as an update from version 7.13 to 7.13.5, include the latest security updates and bug fixes. Minor release upgrades of Red Hat Process Automation Manager, such as an upgrade from version 7.12.x to 7.13, include enhancements, security updates, and bug fixes.



NOTE

Only updates for Red Hat Process Automation Manager are included in Red Hat Process Automation Manager update tools. Updates to Red Hat JBoss EAP must be applied using Red Hat JBoss EAP patch distributions. For more information about Red Hat JBoss EAP patching, see the *Red Hat JBoss EAP patching and upgrading guide* .

Prerequisites

- Your Red Hat Process Automation Manager and KIE Server instances are not running. Do not apply updates while you are running an instance of Red Hat Process Automation Manager or KIE Server.

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.
   If you are upgrading to a new minor release of Red Hat Process Automation Manager, such as an upgrade from version 7.12.x to 7.13, first apply the latest patch update to your current version of Red Hat Process Automation Manager and then follow this procedure again to upgrade to the new minor release.

2. Click **Patches**, download the **Red Hat Process Automation Manager [VERSION] Update Tool** and extract the downloaded **rhpam-$VERSION-update.zip** file to a temporary directory.
   This update tool automates the update of certain components of Red Hat Process Automation Manager, such as Business Central, KIE Server, and the headless Process Automation Manager controller. Use this update tool first to apply updates and then install any other updates or new release artifacts that are relevant to your Red Hat Process Automation Manager distribution.

3. If you want to preserve any files from being updated by the update tool, navigate to the extracted **rhpam-$VERSION-update** folder, open the **blacklist.txt** file, and add the relative paths to the files that you do not want to be updated.
   When a file is listed in the **blacklist.txt** file, the update script does not replace the file with the new version but instead leaves the file in place and in the same location adds the new version with a **.new** suffix. If you block files that are no longer being distributed, the update tool creates an empty marker file with a **.removed** suffix. You can then choose to retain, merge, or delete these new files manually.

Example files to be excluded in **blacklist.txt** file:

```
WEB-INF/web.xml  // Custom file
styles/base.css  // Obsolete custom file kept for record
```

The contents of the blocked file directories after the update:

```
$ ls WEB-INF
web.xml web.xml.new
```

```
$ ls styles
base.css base.css.removed
```

4. In your command terminal, navigate to the temporary directory where you extracted the **rhpam-$VERSION-update.zip** file and run the **apply-updates** script in the following format:

IMPORTANT

Make sure that your Red Hat Process Automation Manager and KIE Server instances are not running before you apply updates. Do not apply updates while you are running an instance of Red Hat Process Automation Manager or KIE Server.

On Linux or Unix-based systems:

```
$ ./apply-updates.sh $DISTRO_PATH $DISTRO_TYPE
```

On Windows:

```
$ .\apply-updates.bat $DISTRO_PATH $DISTRO_TYPE
```

The **$DISTRO_PATH** portion is the path to the relevant distribution directory and the **$DISTRO_TYPE** portion is the type of distribution that you are updating with this update.

The following distribution types are supported in Red Hat Process Automation Manager update tool:

- **rhpam-business-central-eap7-deployable**: Updates Business Central (**business-central.war**)
- **rhpam-kie-server-ee8**: Updates KIE Server (**kie-server.war**)
- **rhpam-kie-server-jws**: Updates KIE Server on Red Hat JBoss Web Server (**kie-server.war**)
- **rhpam-controller-ee7**: Updates the headless Process Automation Manager controller (**controller.war**)
- **rhpam-controller-jws**: Updates the headless Process Automation Manager controller on Red Hat JBoss Web Server (**controller.war**)
  Example update to Business Central and KIE Server for a full Red Hat Process Automation Manager distribution on Red Hat JBoss EAP:

  ```
  $ ./apply-updates.sh ~EAP_HOME/standalone/deployments/business-central.war
  ```

```
rhpam-business-central-eap7-deployable

$ ./apply-updates.sh ~EAP_HOME/standalone/deployments/kie-server.war rhpam-kie-server-ee8
```

Example update to headless Process Automation Manager controller, if used:

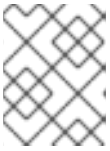
```
$ ./apply-updates.sh ~EAP_HOME/standalone/deployments/controller.war rhpam-controller-ee7
```

The update script creates a **backup** folder in the extracted **rhpam-$VERSION-update** folder with a copy of the specified distribution, and then proceeds with the update.

5. After the update tool completes, return to the **Software Downloads** page of the Red Hat Customer Portal where you downloaded the update tool and install any other updates or new release artifacts that are relevant to your Red Hat Process Automation Manager distribution. For files that already exist in your Red Hat Process Automation Manager distribution, such as **.jar** files for the decision engine or other add-ons, replace the existing version of the file with the new version from the Red Hat Customer Portal.

6. If you use the standalone **Red Hat Process Automation Manager 7.13.5 Maven Repository** artifact (**rhpam-7.13.5-maven-repository.zip**), such as in air-gap environments, download **Red Hat Process Automation Manager 7.13.5 Maven Repository** and extract the downloaded **rhpam-7.13.5-maven-repository.zip** file to your existing **~/maven-repository** directory to update the relevant contents.
Example Maven repository update:

```
$ unzip -o rhpam-7.13.5-maven-repository.zip 'rhba-7.13.5.GA-maven-repository/maven-repository/*' -d /tmp/rhbaMavenRepoUpdate

$ mv /tmp/rhbaMavenRepoUpdate/rhba-7.13.5.GA-maven-repository/maven-repository/ $REPO_PATH/
```



NOTE

You can remove the **/tmp/rhbaMavenRepoUpdate** folder after you complete the update.

7. Optional: If you are changing Red Hat Process Automation Manager from using property-based user storage to file-based user storage, complete the following steps:

   a. Navigate to the **$JBOSS_HOME** directory and run one of the following commands:

   - On Linux or Unix-based systems:

     ```
     $ ./bin/standalone.sh --admin-only -c standalone-full.xml
     ```

     ```
     $ ./bin/jboss-cli.sh --connect --file=rhpam-$VERSION-update/elytron/add-kie-fs-realm.cli
     ```

   - On Windows:

     ```
     $ ./bin/standalone.bat --admin-only -c standalone-full.xml
     ```

```
$ ./bin/jboss-cli.bat --connect --file=rhpam-$VERSION-update/elytron/add-kie-fs-realm.cli
```

b. Run the following command:

- On Linux or Unix-based systems:

```
$ ./bin/elytron-tool.sh filesystem-realm --users-file standalone/configuration/application-users.properties --roles-file standalone/configuration/application-roles.properties --output-location standalone/configuration/kie-fs-realm-users --filesystem-realm-name kie-fs-realm-users
```

- On Windows:

```
$ ./bin/elytron-tool.bat filesystem-realm --users-file standalone/configuration/application-users.properties --roles-file standalone/configuration/application-roles.properties --output-location standalone/configuration/kie-fs-realm-users --filesystem-realm-name kie-fs-realm-users
```

c. Navigate to the directory where you extracted the **rhpam-$VERSION-update.zip** file and run one of the following commands to apply the **kie-fs-realm** patch:

- On Linux or Unix-based systems:

```
$ ./elytron/kie-fs-realm-patch.sh ~/$JBOSS_HOME/standalone/configuration/kie-fs-realm-users/
```

- On Windows:

```
$ ./elytron/kie-fs-realm-patch.bat ~/$JBOSS_HOME/standalone/configuration/kie-fs-realm-users/
```

8. After you finish applying all relevant updates, start Red Hat Process Automation Manager and KIE Server and log in to Business Central.

9. Verify that all project data is present and accurate in Business Central, and in the top-right corner of the Business Central window, click your profile name and click **About** to verify the updated product version number.
If you encounter errors or notice any missing data in Business Central, you can restore the contents in the **backup** folder within the **rhpam-$VERSION-update** folder to revert the update tool changes. You can also reinstall the relevant release artifacts from your previous version of Red Hat Process Automation Manager in the Red Hat Customer Portal. After restoring your previous distribution, you can try again to run the update.

CHAPTER 63. ADDITIONAL RESOURCES

- *Planning a Red Hat Decision Manager installation*
- *Patching and upgrading Red Hat Decision Manager 7.10*
- *Release notes for Red Hat Decision Manager 7.13*

APPENDIX C. VERSIONING INFORMATION

Documentation last updated on Thursday, March 14th, 2024.

APPENDIX D. CONTACT INFORMATION

Red Hat Decision Manager documentation team: brms-docs@redhat.com