



Red Hat Process Automation Manager 7.0

Installing and configuring Red Hat Process Automation Manager in a Red Hat JBoss EAP 7.1 clustered environment

Red Hat Process Automation Manager 7.0 Installing and configuring Red Hat Process Automation Manager in a Red Hat JBoss EAP 7.1 clustered environment

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

Legal Notice

Copyright © 2018 Red Hat, Inc.

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

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

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

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

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

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

This document describes how to create a Red Hat Process Automation Manager clustered environment on Red Hat JBoss Enterprise Application Platform 7.1.

Table of Contents

PREFACE	3
CHAPTER 1. RED HAT PROCESS AUTOMATION MANAGER CLUSTERS	4
CHAPTER 2. DOMAIN MODE CLUSTERS	5
CHAPTER 3. RED HAT PROCESS AUTOMATION MANAGER CLUSTERS IN A DEVELOPMENT (AUTHORING) ENVIRONMENT	6
3.1. INSTALLING AND CONFIGURING ELASTICSEARCH	6
3.2. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.1 AND RED HAT PROCESS AUTOMATION MANAGER	7
3.3. CONFIGURING A RED HAT JBOSS EAP 7.1 DOMAIN-MODE CLUSTER FOR BUSINESS CENTRAL	8
3.4. CONFIGURING RED HAT JBOSS EAP 7.1 HOST FILES FOR A BUSINESS CENTRAL CLUSTERED ENVIRONMENT	10
3.5. STARTING THE CLUSTER AND DISTRIBUTING BUSINESS CENTRAL	12
CHAPTER 4. PROCESS SERVER CLUSTERS IN A RUNTIME ENVIRONMENT	15
4.1. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.1 AND PROCESS SERVER	15
4.2. CONFIGURING A RED HAT JBOSS EAP 7.1 DOMAIN-MODE CLUSTER FOR PROCESS SERVER	16
4.3. CONFIGURING RED HAT JBOSS EAP 7.1 HOST FILES FOR A PROCESS SERVER CLUSTERED ENVIRONMENT	19
4.4. STARTING THE CLUSTER AND DISTRIBUTING PROCESS SERVER	20
4.5. CLUSTERING WITH THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER	21
4.6. CLUSTERING SMART ROUTER FOR A RUNTIME ENVIRONMENT	23
CHAPTER 5. RELATED INFORMATION	26
APPENDIX A. VERSIONING INFORMATION	27

PREFACE

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.

Prerequisite

You have reviewed the information in [Planning a Red Hat Process Automation Manager installation](#).

CHAPTER 1. RED HAT PROCESS AUTOMATION 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 a loss of data when a single computer fails because data on all of the computers (nodes) of the cluster is replicated on each node of the cluster. 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. Load balancing shares the computing load across the nodes of the cluster. Doing this improves the overall performance.

There are several ways that you can cluster Red Hat Process Automation Manager components. This document describes how to cluster the following scenarios in domain mode:

- [Chapter 3, *Red Hat Process Automation Manager clusters in a development \(authoring\) environment*](#)
- [Chapter 4, *Process Server clusters in a runtime environment*](#)

CHAPTER 2. DOMAIN MODE CLUSTERS

By using domain mode with a cluster, you can centrally store, manage, and publish configurations for your servers. This section explains key concepts in domain mode clustering.

Domain controller: The domain controller is responsible for storing, managing, and publishing the general configuration for each node in the cluster. It is the central point from which nodes in a cluster obtain their configuration. The node that runs the domain controller is referred to as the master node and the other nodes of the cluster are referred to as slave nodes. The domain controller is configured with the **domain.xml** file.

Host controller: The host controller manages server instances on a specific node. You configure it to run one or more server instances. The domain controller can also interact with the host controllers on each system to manage the cluster. To reduce the number of running processes, a domain controller also acts as a host controller on the system it runs on. On slave nodes, the host controller is configured with the **host.xml** file.

Domain profile: A domain profile is a named configuration that a server can use to boot from. A domain controller can define multiple domain profiles that are consumed by different servers.

Server group: A server group is a set of server instances that are managed and configured together. You can manage configurations, deployments, socket bindings, modules, extensions, and system properties for each server group. You can assign a domain profile to a server group and every service in that group will use that domain profile as their configuration.

In domain mode, a domain controller is started on the master node of the cluster. The configuration for the cluster is located in the domain controller. Next, a host controller is started on all of slave nodes of the cluster. Each host controller deployment configuration specifies how many server instances will be started on that system. When the host controller boots up, it starts as many server instances as it was configured to do. These server instances pull their configuration from the domain controller.

CHAPTER 3. RED HAT PROCESS AUTOMATION MANAGER CLUSTERS IN A DEVELOPMENT (AUTHORING) ENVIRONMENT

Developers use Red Hat Process Automation Manager to author rules and processes that assist users with decision making. The primary benefit of clustering Red Hat Process Automation Manager development environments is high availability. If clustering is configured and Developer X is working on Node1 and that node fails, Developer X's work is preserved and visible on any other node of the cluster.

Most development environments consist of Business Central for creating rules and processes and at least one Process Server to test those rules and processes .



NOTE

You can also use this configuration for runtime environments if you want to use Business Central to manage the Process Servers in the cluster.

To create a Red Hat Process Automation Manager clustered development environment, you download and extract Red Hat JBoss EAP 7.1 and Red Hat Process Automation Manager 7.0. Then, you configure Red Hat JBoss EAP 7.1 for a domain-mode cluster, including Elasticsearch and the Artemis Java messaging server (JMS) broker resource adapter integrated with Red Hat JBoss EAP. Finally, you start the cluster and install Business Central on each cluster node.

Elasticsearch is a highly scalable open source full-text search and analytics engine. It enables you to store, search, and analyze high volumes of data quickly and in near-real time. In a Red Hat Process Automation Manager clustered environment, it enables you to perform complex and efficient searches across cluster nodes.

The Artemis JMS broker resource adapter 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.

3.1. INSTALLING AND CONFIGURING ELASTICSEARCH

Elasticsearch is a highly scalable open source full-text search and analytics engine. It enables you to store, search, and analyze high volumes of data quickly and in near-real time. In a Red Hat Process Automation Manager clustered environment, it enables you to perform complex and efficient searches across cluster nodes.

This procedure describes how to set up Elasticsearch for a simplified, non-high availability environment. To configure Elasticsearch for a high availability Red Hat Process Automation Manager clustered environment, see the [Elasticsearch documentation](#).

Procedure

1. Download and unzip the [Elasticsearch](#) installation file.
2. Replace the contents of the `elasticsearch/config/elasticsearch.yml` file with the following content, where `<ELASTICSEARCH_NODE_IP>` is the IP address of the system where you unzipped the Elasticsearch file:

```
cluster.name: kie-cluster
transport.host: <ELASTICSEARCH_NODE_IP>
```

```
http.host: <ELASTICSEARCH_NODE_IP>
transport.tcp.port: 9300
discovery.zen.minimum_master_nodes: 1
```

3. Run Elasticsearch:

- On Linux or UNIX-based systems, enter:

```
ELASTICSEARCH_HOME/bin/elasticsearch
```

On Windows, enter:

```
ELASTICSEARCH_HOME\bin\elasticsearch.bat
```

3.2. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.1 AND RED HAT PROCESS AUTOMATION MANAGER

Complete the steps in this section to download and install Red Hat JBoss EAP 7.1 and download and repackage Red Hat Process Automation Manager 7.0 for installation in a clustered environment.

Procedure

1. Install Red Hat JBoss EAP 7.1 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: 7.1**
 - b. Click **Download** next to **Red Hat JBoss Enterprise Application Platform 7.1.0. (JBEAP-7.1.0/jboss-eap-7.1.0.zip)**.
 - c. Extract the **jboss-eap-7.1.0.zip** file. In the following steps, **EAP_HOME** is the **jboss-eap-7.1/jboss-eap-7.1** directory.
2. On the cluster node where the Red Hat JBoss EAP domain controller will be started (the master node), 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: 7.0**
3. Download **Red Hat Process Automation Manager 7.0.0 Business Central Deployable for Red Hat JBoss EAP 7 (rhpam-7.0.0-business-central-eap7-deployable.zip)**.
4. Extract the **rhpam-7.0.0-business-central-eap7-deployable.zip** file to a temporary directory. In the following examples this directory is called **TEMP_DIR**.
5. Copy the contents of **TEMP_DIR/rhpam-7.0.0-business-central-eap7-deployable/jboss-eap-7.1** to **EAP_HOME**.
6. Repackage the **business-central.war** file:

- a. Navigate to the **`TEMP_DIR/rhpam-7.0.0-business-central-eap7-deployable/jboss-eap-7.1/standalone/deployments/business-central.war`** directory.
- b. Select the contents of the **`TEMP_DIR/rhpam-7.0.0-business-central-eap7-deployable/jboss-eap-7.1/standalone/deployments/business-central.war`** directory and create the **`business-central.zip`** file.
- c. Rename **`business-central.zip`** to **`business-central.war`**. This is the file that you will use to deploy Business Central on the cluster nodes.
- d. If desired, copy the new **`business-central.war`** file to a location that is more convenient to deploy from.

3.3. CONFIGURING A RED HAT JBOSS EAP 7.1 DOMAIN-MODE CLUSTER FOR BUSINESS CENTRAL

By using domain mode with a cluster, you can centrally manage and publish configurations for your servers. Domain mode provides a central location to store and publish configurations. Complete the steps in this section to configure a Red Hat JBoss EAP 7.1 cluster in domain mode, with Elasticsearch and the Artemis Java messaging server (JMS) broker resource adapter integrated with Red Hat JBoss EAP.



NOTE

These steps describe a basic cluster configuration. For more complex configurations, see the [Red Hat JBoss EAP 7.1 Configuration Guide](#).

Prerequisites

- Elasticsearch is installed as described in [Section 3.1, “Installing and configuring Elasticsearch”](#).
- Red Hat JBoss EAP is installed on each node of the cluster as described in [Section 3.2, “Downloading and extracting Red Hat JBoss EAP 7.1 and Red Hat Process Automation Manager”](#).
- An NFS server with a mounted partition is available.

Procedure

1. On the the master node of the cluster (where you will run the domain controller), open the **`EAP_HOME/domain/configuration/domain.xml`** file in a text editor.
2. Make the following changes for each server group in the **`server-groups`** element:
 - a. Specify the **`server-group name`** and **`jvm`** name.
 - b. To enable high availability, change the **`profile`** to **`"full-ha"`** and the **`socket-binding-group`** to **`"full-ha-sockets"`**.
 - c. To prevent Business Central from running out of memory, increase the JVM maximum heap size to **`1500m`**.

```
<server-group name="some-server-group" profile="full-ha">
```

```

    <jvm name="some-jvm">
      <heap size="1000m" max-size="1500m"/>
    </jvm>
    <socket-binding-group ref="full-ha-sockets"/>
  </server-group>

```

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

- **<ELASTICSEARCH_NODE_IP>** is the IP addresses where Elasticsearch is installed.
- **<JMS_USER>** is a Red Hat JBoss EAP application user. It is the user that Business Central uses to connect to the embedded JMS broker.
- **<JMS_USER_PASSWORD>** is the password for **<JMS_USER>**.

```

<system-properties>
  <property name="appformer-jms-connection-mode" value="JNDI"/>
  <property name="appformer-jms-username" value="<JMS_USER>" />
  <property name="appformer-jms-password" value="
<JMS_USER_PASSWORD>"/>
  <property name="org.appformer.ext.metadata.elastic.port"
value="9300"/>
  <property name="org.appformer.ext.metadata.elastic.host"
value="<ELASTICSEARCH_NODE_IP>"/>
  <property name="org.appformer.ext.metadata.elastic.cluster"
value="kie-cluster"/>
  <property name="org.appformer.ext.metadata.index"
value="elastic"/>
  <property name="org.appformer.ext.metadata.elastic.retries"
value="10"/>
  <property name="appformer-jms-connection-factory"
value="java:/ConnectionFactory"/>
</system-properties>

```

4. In the **<subsystem xmlns="urn:jboss:domain:messaging-activemq:2.0">** element, make the following changes:

- a. Replace **<cluster password="\${jboss.messaging.cluster.password:CHANGE ME!}" />** with **<cluster user="<CLUSTER_USER>" password="<CLUSTER_PWD>" />** where **<CLUSTER_USER>** is the user used by Red Hat JBoss EAP to form a messaging cluster and **<CLUSTER_PWD>** is the password for that user.
- b. In the **security-setting** element, add the role of the **<JMS_USER>** to a new **role-name** element, for example:

```

<security-setting name="#">
  <role name="guest" send="true" consume="true" create-non-
durable-queue="true" delete-non-durable-queue="true"/>
  <role name="admin" send="true" consume="true" create-non-
durable-queue="true" delete-non-durable-queue="true"/>
</security-setting>

```

5. Save the **domain.xml** file.

3.4. CONFIGURING RED HAT JBOSS EAP 7.1 HOST FILES FOR A BUSINESS CENTRAL CLUSTERED ENVIRONMENT

After you have installed Red Hat JBoss EAP and configured the `domain.xml` file for a Business Central clustered environment, complete these steps in the `host.xml` file on the master node and in the `host-slave.xml` file on each slave node.

Prerequisite

Red Hat JBoss EAP 7.1 is installed on each node of the cluster.

Procedure

1. Open the `EAP_HOME/domain/configuration/host.xml` file or the `EAP_HOME/domain/configuration/host-slave.xml` file in a text editor.
2. In the `main-server-group <servers>` element, add the servers that will be part of the cluster.
3. Add the the system properties listed in [Table 3.1, “Cluster node properties”](#)
4. Save the `host.xml` or `host-slave.xml` file.

Table 3.1. Cluster node properties

Property name	Value	Description
<code>jboss.node.name</code>	<i>node name</i>	A node name unique within the cluster (optional).
<code>org.uberfire.nio.git.daemon.host</code>	<i>node name</i>	The name of the daemon host machine in a physical cluster.
<code>org.uberfire.nio.git.daemon.port</code>	<i>port number</i>	The port used by the VFS repo to accept client connections. The port must be unique for each cluster member.
<code>org.uberfire.nio.git.ssh.host</code>	<i>SSH host node name</i>	The name of the SSH host machine in a physical cluster.
<code>org.uberfire.nio.git.ssh.port</code>	<i>port number</i>	The unique port number for ssh access to the GIT repo for a cluster running on physical machines.
<code>org.uberfire.nio.git.dir</code>	<i>.niogit directory</i>	The location of Git repositories stored under .niogit , on an NFS mounted partition accessible to a Red Hat JBoss EAP user.
<code>org.uberfire.nio.git.ssh.hostport</code> and <code>org.uberfire.nio.git.daemon.hostport</code>	<i>SSH host port and daemon host port</i>	In a virtualized environment, the outside port to be used.

Property name	Value	Description
org.uberfire.nio.git.ssh.cert.dir	<i>.security directory</i>	The location of the .security directory where local certificates will be stored.

The following examples demonstrate how to configure a three node cluster in the **host.xml** and **host-slave.xml** files:

Example 3.1. Cluster nodeOne Configuration

```
<system-properties>
  <property name="jboss.node.name" value="nodeOne"/>
  <property name="org.uberfire.nio.git.daemon.host" value="nodeOne"/>
  <property name="org.uberfire.nio.git.daemon.port" value="9418" boot-
time="false"/>
  <property name="org.uberfire.nio.git.daemon.hostport" value="9418"
boot-time="false"/>
  <property name="org.uberfire.nio.git.ssh.host" value="nodeOne"/>
  <property name="org.uberfire.nio.git.ssh.port" value="8003" boot-
time="false"/>
  <property name="org.uberfire.nio.git.ssh.hostport" value="8003" boot-
time="false"/>
  <property name="org.uberfire.nio.git.dir" value="/mnt/jbpm/niogit"
boot-time="false"/>
  <property name="org.uberfire.nio.git.ssh.cert.dir"
value="/tmp/jbpm/nodeone"
boot-time="false"/>
</system-properties>
```

Example 3.2. Cluster nodeTwo Configuration

```
<system-properties>
  <property name="jboss.node.name" value="nodeTwo"/>
  <property name="org.uberfire.nio.git.daemon.host" value="nodeTwo" />
  <property name="org.uberfire.nio.git.daemon.port" value="9419" boot-
time="false"/>
  <property name="org.uberfire.nio.git.daemon.hostport" value="9419"
boot-time="false"/>
  <property name="org.uberfire.nio.git.ssh.host" value="nodeTwo" />
  <property name="org.uberfire.nio.git.ssh.port" value="8004" boot-
time="false"/>
  <property name="org.uberfire.nio.git.ssh.hostport" value="8004" boot-
time="false"/>
  <property name="org.uberfire.nio.git.dir" value="/mnt/jbpm/niogit"
boot-time="false"/>
  <property name="org.uberfire.nio.git.ssh.cert.dir"
value="/tmp/jbpm/nodetwo" boot-time="false"/>
</system-properties>
```

Example 3.3. Cluster nodeThree Configuration

```

<system-properties>
  <property name="jboss.node.name" value="nodeThree" />
  <property name="org.uberfire.nio.git.daemon.host" value="nodeThree" />
  <property name="org.uberfire.nio.git.daemon.port" value="9420" boot-
time="false"/>
  <property name="org.uberfire.nio.git.daemon.hostport" value="9420"
    boot-time="false"/>
  <property name="org.uberfire.nio.git.ssh.host" value="nodeThree" />
  <property name="org.uberfire.nio.git.ssh.port" value="8005" boot-
time="false"/>
  <property name="org.uberfire.nio.git.ssh.hostport" value="8005" boot-
time="false"/>
  <property name="org.uberfire.nio.git.dir" value="/mnt/jbpm/niogit"
boot-time="false"/>
  <property name="org.uberfire.nio.git.ssh.cert.dir"
value="/tmp/jbpm/nodethree"
    boot-time="false"/>
</system-properties>

```

3.5. STARTING THE CLUSTER AND DISTRIBUTING BUSINESS CENTRAL

After you have configured a Red Hat JBoss EAP cluster and repackaged Business Central, you must start the cluster and distribute the repackaged **business-central.war** file.

Prerequisites

- Elasticsearch is installed and running on one node of the cluster.
- A Red Hat JBoss EAP 7.1 cluster is available and the **domain.xml**, **host.xml**, and **host-slave.xml** files are configured.
- The repackaged **business-central.war** file is available.

Procedure

1. Add Red Hat JBoss EAP management users to the master node of the cluster (where you configured the **domain.xml** file) as described in the [Red Hat JBoss EAP 7.1 Configuration Guide](#).
2. On each node of the cluster, add the following users:
 - a. Add Business Central administration users as described in [Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1](#).
 - b. Add the JMS user, where **<JMS_USER>** and **<JMS_PWD>** is the JMS user and password combination that you added to the **domain.xml** file:

```

$ ./add-user.sh -a --user <JMS_USER> --password <JMS_PWD> --role
admin

```


3. Start the cluster by starting the the master node first, then start each of the slave nodes.

a. To start the master node, enter one of the following commands:

- On Linux or UNIX-based systems, enter:

```
<EAP_HOME>/bin/domain.sh
```

- On Windows, enter:

```
<EAP_HOME>/bin/domain.bat
```

b. To start the slave nodes, on each of the slave nodes enter one of the following commands, where **<MASTER_NODE_IP>** is the IP address of the master node (where the domain controller is configured):

- On Linux or UNIX-based systems, enter:

```
<EAP_HOME>/bin/domain.sh --host-config=host-slave.xml -  
Djboss.domain.master.address=<MASTER_NODE_IP>
```

- On Windows, enter:

```
<EAP_HOME>/bin/domain.bat --host-config=host-slave.xml -  
Djboss.domain.master.address=<MASTER_NODE_IP>
```

For more information, see the following sections of the *Red Hat JBoss EAP 7.1 Configuration Guide*:

- ["Starting JBoss EAP"](#)
- ["Managed Domain Configuration Files"](#)

4. On each node of the cluster, review the

***EAP_HOME*/domain/servers/SERVER_NAME/log/server.log** file to verify that the nodes are available.

5. To deploy the **business-central.war** file into the server group, complete the following steps:

- On the master node, log in to the Red Hat JBoss EAP **Administration** console as a **management** user.
- Click **Deployments** → **Server Groups** → **main-server-group** and click **Add**.
- In the dialog box, click **Upload a new deployment** and click **Next**.
- In the **Upload Deployments** dialog box, click **Browse**, select the **business-central.war** file, and click **Next**.
- Click **Enable** and click **Next**. When the deployment is complete, Business Central starts automatically on each node.



NOTE

Make sure to check deployment unit readiness with every cluster member.

When a deployment unit is created on a cluster node, it takes some time before it is distributed among all cluster members. You can check the deployment status using the server Administration console or REST API. However, if the query is sent to the node where the deployment was originally issued, the query will return a value of **deployed**. If the query is sent to a node where the deployment has not yet been distributed, the query returns **DeploymentNotFoundException**.

For more information about installing Business Central, see [Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1](#).

CHAPTER 4. PROCESS SERVER CLUSTERS IN A RUNTIME ENVIRONMENT

In a runtime environment, Process Server runs rules and processes that support business decisions. The primary benefit of clustering a Process 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 Process Server clustered runtime environment, you download and extract Red Hat JBoss EAP 7.1 and Process Server. Then, you configure Red Hat JBoss EAP 7.1 for a domain-mode cluster, start the cluster, and install Process Server on each cluster node.

If desired, you can then cluster the headless Process Automation Manager controller and Smart Router.

4.1. DOWNLOADING AND EXTRACTING RED HAT JBOSS EAP 7.1 AND PROCESS SERVER

Complete the steps in this section to download and install Red Hat JBoss EAP 7.1 and download and repackage Process Server for installation in a clustered environment.

Procedure

1. Install Red Hat JBoss EAP 7.1 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: 7.1**
 - b. Click **Download** next to **Red Hat JBoss Enterprise Application Platform 7.1.0**. (**jboss-eap-7.1.0.zip**).
 - c. Extract the **jboss-eap-7.1.0.zip** file. The **jboss-eap-7.1/jboss-eap-7.1** directory is referred to as **EAP_HOME**.
2. Download and extract Process 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: 7.0**
 - Download **Red Hat Process Automation Manager 7.0.0 Process Server for All Supported EE7 Containers** (**rhpm-7.0.0-kie-server-ee7.zip**).
 - b. Extract the **rhpm-7.0.0-kie-server-ee7.zip** archive to a temporary directory. In the following examples this directory is called **TEMP_DIR**.
3. Repackage the **kie-server.war** directory:

- a. Navigate to the **`TEMP_DIR/rhpam-7.0.0-kie-server-ee7/kie-server.war`** directory.
 - b. Select the contents of the **`TEMP_DIR/rhpam-7.0.0-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 Process Server on the cluster nodes.
 - d. If desired, copy the new **`kie-server.war`** file to a location that is more convenient to deploy from.
4. If you want to use a security manager with Process Server, copy the contents of the **`TEMP_DIR/rhpam-7.0.0-kie-server-ee7/SecurityPolicy`** directory to the **`EAP_HOME/bin`** directory on each node of the cluster.

4.2. CONFIGURING A RED HAT JBOSS EAP 7.1 DOMAIN-MODE CLUSTER FOR PROCESS SERVER

By using domain mode with a cluster, you can centrally manage and publish configurations for your servers. Domain mode provides a central location to store and publish configurations. Complete the steps in this section to configure a Red Hat JBoss EAP cluster in domain mode to use with Process Server.

Procedure

1. Install the JDBC driver on all Red Hat JBoss EAP instances that are part of this cluster. For more information, see the ["JDBC Drivers"](#) section of the *Red Hat JBoss EAP 7.1 Configuration Guide*.
2. On the master node of the cluster (where you will run the domain controller), open the **`EAP_HOME/domain/configuration/domain.xml`** file in a text editor.
3. Make the following changes for each server group in the **`server-groups`** element:
 - a. Specify the **`server-group name`** and **`jvm name`**.
 - b. To enable high availability, change the **`profile`** to **`full-ha`** and the **`socket-binding-group`** to **`full-ha-sockets`**.

```
<server-group name="some-server-group" profile="full-ha">
  <jvm name="some-jvm">
    <heap size="1000m" max-size="1000m"/>
  </jvm>
  <socket-binding-group ref="full-ha-sockets"/>
</server-group>
```

4. Edit the **`data-stores`** property and the **`timer-service thread-pool-name`** above it:
 - The **`datasource-jndi-name`** is the JNDI name of the database specified at the beginning of this procedure.
 - You can enter any name for the value of the **`partition`** property. However, a node will only see timers from other nodes that have the same partition name. Grouping nodes into partitions by assigning partition names enables you to break a large cluster up into several

smaller clusters. Doing this should improve performance. For example, instead of having a cluster of 100 nodes, where all 100 nodes are trying to execute and refresh the same timers, you can create 20 clusters of 5 nodes by giving every group of 5 a different partition name.

- Replace the **default-data-store** attribute value with **ejb_timer_ds**.
- Set the value of **refresh-interval** in milliseconds to specify how often the EJB timer connects to the database to synchronize and load tasks to be processed.

```
<timer-service thread-pool-name="default" default-data-
store="ejb_timer_ds">
  <data-stores>
    <database-data-store name="ejb_timer_ds" datasource-jndi-
name="java:jboss/datasources/ejb_timer" database="postgresql"
partition="ejb_timer_part" refresh-interval="30000"/>
  </data-stores>
</timer-service>
```

The following table lists the supported databases and the corresponding **database** attribute value:

Database	Attribute value
Hyper SQL (for demonstration purposes only, not supported)	hsq1
PostgreSQL	postgresql
Oracle	oracle
IBM DB2	db2
Microsoft SQL Server	mssql
MySQL and MariaDB	mysql

5. Add the Process Server and EJB timer data sources to the **full-ha** profile. In these examples, **<DATABASE>** is the name of the database, **<SERVER_NAME>** is the host name of the JNDI database, and **<USER_NAME>** and **<USER_PWD>** are the credentials for that database.

- Add the data source to allow Process Server to connect to the database, for example:

```
<xa-datasource jndi-name="java:/jboss/datasources/rhpam" pool-
name="rhpam-RHPAM" use-java-context="true" enabled="true">
  <xa-datasource-property name="DatabaseName"><DATABASE></xa-
datasource-property>
  <xa-datasource-property name="PortNumber">5432</xa-datasource-
property>
  <xa-datasource-property name="ServerName"><SERVER_NAME></xa-
datasource-property>
  <driver>postgresql</driver>
  <security>
```

```

        <user-name><USER_NAME></user-name>
        <password><USER_PWD></password>
    </security>
</xa-datasource>

```

- Add the data source to enable the EJB timer, for example:

```

<xa-datasource jndi-name="java:jboss/datasources/ejb_timer" pool-
name="ejb_timer" use-java-context="true" enabled="true">
    <xa-datasource-property name="DatabaseName"><DATABASE></xa-
datasource-property>
    <xa-datasource-property name="PortNumber">5432</xa-
datasource-property>
    <xa-datasource-property name="ServerName"><SERVER_NAME></xa-
datasource-property>
    <driver>postgresql</driver>
    <transaction-
isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
    <security>
        <user-name><USER_NAME></user-name>
        <password><USER_PWD></password>
    </security>
</xa-datasource>

```



WARNING

You must use two different databases for Process Server runtime data and EJB timer data.

6. In the `<subsystem xmlns="urn:jboss:domain:messaging-activemq:2.0">` element, make the following changes:
 - a. Replace `<cluster password="${jboss.messaging.cluster.password:CHANGE ME!!}" />` with `<cluster user="<CLUSTER_USER>" password="<CLUSTER_PWD>" />` where `<CLUSTER_USER>` is the user used by Red Hat JBoss EAP to form a messaging cluster and `<CLUSTER_PWD>` is the password for that user.
 - b. In the `security-setting` element add the `kie-server` role:

```

<subsystem xmlns="urn:jboss:domain:messaging-activemq:2.0">
    <server name="default">
        <security-setting name="#">
            <role name="guest" send="true" consume="true"
create-non-durable-queue="true" delete-non-durable-queue="true"/>
            <role name="kie-server" send="true" consume="true"
create-non-durable-queue="true" delete-non-durable-queue="true"/>
        </security-setting>
    </server>
</subsystem>

```

7. Add the following properties to the `<system-properties>` element and replace the following placeholders:

- **<JNDI_NAME>** is the JNDI name of your data source. For Red Hat Process Automation Manager, this is **java:/jboss/datasources/rhpam**.
- **<DIALECT>** is the hibernate dialect for your database.
The following dialects are supported:
 - DB2: **org.hibernate.dialect.DB2Dialect**
 - MSSQL: **org.hibernate.dialect.SQLServer2012Dialect**
 - MySQL: **org.hibernate.dialect.MySQL5InnoDBDialect**
 - MariaDB: **org.hibernate.dialect.MySQL5InnoDBDialect**
 - Oracle: **org.hibernate.dialect.Oracle10gDialect**
 - PostgreSQL: **org.hibernate.dialect.PostgreSQL82Dialect**
 - PostgreSQL plus: **org.hibernate.dialect.PostgresPlusDialect**

```
<system-properties>
  <property name="org.kie.server.persistence.ds" value="
<JNDI_NAME>"/>
  <property name="org.kie.server.persistence.dialect" value="
<DIALECT>"/>
  <property name="org.jbpm.ejb.timer.tx" value="true"/>
</system-properties>
```

8. Save the **domain.xml** file.

4.3. CONFIGURING RED HAT JBOSS EAP 7.1 HOST FILES FOR A PROCESS SERVER CLUSTERED ENVIRONMENT

After you have installed Red Hat JBoss EAP and configured the **domain.xml** file for a Process Server clustered environment, you must configure the **host.xml** and **host-slave.xml** files.

Prerequisite

Red Hat JBoss EAP 7.1 is installed on each node of the cluster.

Procedure

1. Complete the following steps on the master node of the cluster:
 - a. Open the **EAP_HOME/domain/configuration/host.xml** file in a text editor.
 - b. In the **main-server-group <servers>** element, add the servers that will be part of the cluster.
2. Complete the following steps in the **host.xml** file on the master node and in the **host-slave.xml** file on each slave node:
 - a. Open the **EAP_HOME/domain/configuration/host-slave.xml** file in a text editor.

- b. In the **main-server-group** <servers> element, add the servers that will be part of the cluster.

4.4. STARTING THE CLUSTER AND DISTRIBUTING PROCESS SERVER

After you have configured a Red Hat JBoss EAP cluster and repackaged Process Server, you must start the cluster and distribute the repackaged **kie-server.war** file.

Prerequisites

- A Red Hat JBoss EAP 7.1 cluster is available and the **domain.xml**, **host.xml**, and **host-slave.xml** files have been configured.
- The repackaged **kie-server.war** file is available.

Procedure

1. Add Red Hat JBoss EAP management users to the master node (where you configured the **domain.xml** file) as described in the [Red Hat JBoss EAP 7.1 Configuration Guide](#).
2. On each node of the cluster, add Process Server users as described in [Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1](#).

- a. To start the master node, enter one of the following commands:

- On Linux or UNIX-based systems, enter:

```
<EAP_HOME>/bin/domain.sh
```

On Windows, enter:

```
<EAP_HOME>/bin/domain.bat
```

- b. To start the slave nodes, on each of the slave nodes enter one of the following commands, where **<MASTER_NODE_IP>** is the IP address of the master node (where the domain controller is configured):

- On Linux or UNIX-based systems, enter:

```
<EAP_HOME>/bin/domain.sh --host-config=host-slave.xml -  
Djboss.domain.master.address=<MASTER_NODE_IP>
```

- On Windows, enter:

```
<EAP_HOME>/bin/domain.bat --host-config=host-slave.xml -  
Djboss.domain.master.address=<MASTER_NODE_IP>
```

For more information, see the following sections of the *Red Hat JBoss EAP 7.1 Configuration Guide*:

- ["Starting JBoss EAP"](#)
- ["Managed Domain Configuration Files"](#).

3. On each node of the cluster, review the **`EAP_HOME/domain/servers/SERVER_NAME/log/server.log`** file to verify that the nodes are available.
4. To deploy the **`kie-server.war`** file that you created previously into the server group, complete the following steps on the master node:
 - a. Log in to the Red Hat JBoss EAP **Administration** console of your domain as a **management** user.
 - b. Click **Deployments** → **Server Groups** → **main-server-group** and click **Add**.
 - c. In the dialog box, click **Upload a new deployment** and click **Next**.
 - d. In the **Upload Deployments** dialog box, click **Browse**, select the **`kie-server.war`** file, and click **Next**.
 - e. Click **Enable** and click **Next**.



NOTE

Make sure to check deployment unit readiness with every cluster member.

When a deployment unit is created on a cluster node, it takes some time before it is distributed among all cluster members. You can check the deployment status using the server Administration console or REST API. However, if the query is sent to the node where the deployment was originally issued, the query will return a value of **deployed**. If the query is sent to a node where the deployment has not yet been distributed, the query returns **DeploymentNotFoundException**.

For more information about installing Business Central, see [Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1](#).

4.5. CLUSTERING WITH THE HEADLESS PROCESS AUTOMATION MANAGER CONTROLLER

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

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 Process Server Java Client API to interact with it.

Prerequisites

- A backed-up Red Hat JBoss EAP installation version 7.1 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 mounted partition is available.

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: 7.0**
2. Download **Red Hat Process Automation Manager 7.0.0 Add Ons** (the **rhcam-7.0.0-add-ons.zip** file).
3. Unzip the **rhcam-7.0.0-add-ons.zip** file. The **rhcam-7.0-controller-ee7.zip** file is in the unzipped directory.
4. Extract the **rhcam-7.0-controller-ee7.zip** file to a temporary directory. In the following examples this directory is called **TEMP_DIR**.
5. Repackage the **controller.war** directory:
 - a. Navigate to the **TEMP_DIR/rhcam-7.0.0-add-ons/rhcam-7.0-controller-ee7/controller.war** directory.
 - b. Select the contents of the **TEMP_DIR/rhcam-7.0.0-add-ons/rhcam-7.0-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 controller on the cluster nodes.
 - d. If desired, copy the new **controller.war** file to a location that is more convenient to deploy from.
6. If you want to use a security manager with the controller, copy the contents of the **TEMP_DIR/rhcam-7.0.0-add-ons/rhcam-7.0-controller-ee7/SecurityPolicy** directory to the **EAP_HOME/bin** directory on each node of the cluster.
7. Add Red Hat JBoss EAP management users to the master node (where you configured the **domain.xml** file) as described in the [Red Hat JBoss EAP 7.1 Configuration Guide](#).
8. On each node of the cluster, add users for the headless Process Automation Manager controller as described in the "Installing the headless Process Automation Manager controller" section of the [Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1](#).
9. Complete the following steps in the **host.xml** file on the master node and in the **host-slave.xml** file on each slave node:
 - a. Open the **EAP_HOME/domain/configuration/host.xml** or **EAP_HOME/domain/configuration/host-slave.xml** file in a text editor.
 - b. In the **main-server-group <servers>** element, add the servers that will be part of the cluster.
 - c. 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>
```

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.

10. To deploy the `controller.war` file that you created previously into the server group, complete the following steps on the master node:
 - a. Log in to the Red Hat JBoss EAP **Administration** console of your domain as a **management** user.
 - b. Click **Deployments** → **Server Groups** → **main-server-group** and click **Add**.
 - c. In the dialog box, click **Upload a new deployment** and click **Next**.
 - d. In the **Upload Deployments** dialog box, click **Browse**, select the `controller.war` file, and click **Next**.
 - e. Click **Enable** and click **Next**.



NOTE

Make sure to check deployment unit readiness with every cluster member.

When a deployment unit is created on a cluster node, it takes some time before it is distributed among all cluster members. You can check the deployment status using the server Administration console or REST API. However, if the query is sent to the node where the deployment was originally issued, the query will return a value of **deployed**. If the query is sent to a node where the deployment has not yet been distributed, the query returns **DeploymentNotFoundException**.

For more information about installing Business Central, see [Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1](#).

4.6. CLUSTERING SMART ROUTER FOR A RUNTIME ENVIRONMENT

Smart Router enables you to aggregate multiple independent Process Server instances as though they are a single server. It performs the role of an intelligent load balancer because it can both route requests to individual Process Servers and aggregate data from different Process Servers. Through aliases, Smart Router is a proxy to Process Servers. Smart Router performs the following tasks:

- Collects information from various server instances in a single client request

- Finds the right server for a specific request
- Aggregates responses from different servers
- Provides efficient load-balancing
- Manages changing environments, for example adding and removing server instances
- Manages registration with the Process Server controller

This section describes how to install Smart Router and configure it for a Red Hat Process Automation Manager runtime environment.



NOTE

Load balancing requests for Smart router cluster must be managed externally, using standard load balancing tools.

Prerequisite

Process Server is installed on each node of a Red Hat JBoss EAP 7.1 cluster.

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: 7.0**
2. Download **Red Hat Process Automation Manager 7.0.0 Add-Ons**.
3. Extract the downloaded **rhpm-7.0.0-add-ons.zip** file to a temporary directory. The **rhpm-7.0-smart-router.jar** file is in the extracted **rhpm-7.0.0-add-ons** directory.
4. Copy the **rhpm-7.0-smart-router.jar** file to location where you will run the file.
5. From the temporary directory, enter the following command to start SmartRouter:

```
java
-Dorg.kie.server.router.host=<ROUTER_HOST>
-Dorg.kie.server.router.port=<ROUTER_PORT>
-Dorg.kie.server.controller=<CONTROLLER_URL>
-Dorg.kie.server.controller.user=<CONTROLLER_USER>
-Dorg.kie.server.controller.pwd=<CONTROLLER_PWD>
-Dorg.kie.server.router.config.watcher.enabled=true
-Dorg.kie.server.router.repo=<NFS_STORAGE>
-jar rhpm-7.0-smart-router.jar
```

The properties in the preceding command have the following default values:

```
org.kie.server.router.host=localhost
org.kie.server.router.port=9000
org.kie.server.controller= N/A
org.kie.server.controller.user=kieserver
```

```
org.kie.server.controller.pwd=kieserver1!  
org.kie.server.router.repo= <CURRENT_WORKING_DIR>  
org.kie.server.router.config.watcher.enabled=false
```



NOTE

The router is capable of providing an aggregate sort, however the data returned when used through the management console is in raw format. Therefore, sorting is in whatever way individual servers return.

Paging is supported in its standard format.

6. To use Smart Router from the client side, use the Smart Router URL instead of the Process Server URL, for example:

```
KieServicesConfiguration config =  
KieServicesFactory.newRestConfiguration("http://smartrouter.example.  
com:9000", "USERNAME", "PASSWORD");
```

In this example, **smartrouter.example.com** is the Smart Router URL, and **USERNAME** and **PASSWORD** are the log in credentials for the Smart Router configuration.

CHAPTER 5. RELATED INFORMATION

- [*Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1*](#)
- [*Planning a Red Hat Process Automation Manager installation*](#)
- [*Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1*](#)
- [*Deploying a Red Hat Process Automation Manager authoring environment on Red Hat OpenShift Container Platform*](#)

APPENDIX A. VERSIONING INFORMATION

Documentation last updated on: Monday, October 1, 2018.