Red Hat Process Automation Manager 7.5

Deploying a Red Hat Process Automation Manager authoring environment on Red Hat OpenShift Container Platform

Last Updated: 2021-07-15

Red Hat Process Automation Manager 7.5 Deploying a Red Hat Process Automation Manager authoring environment on Red Hat OpenShift Container Platform

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

Legal Notice

Abstract

This document describes how to deploy a Red Hat Process Automation Manager 7.5 authoring environment on Red Hat OpenShift Container Platform.

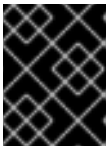
Table of Contents

PREFACE

As a system engineer, you can deploy a Red Hat Process Automation Manager authoring environment on Red Hat OpenShift Container Platform to provide a platform for development of services, process applications, and other business assets.

Prerequisites

- Red Hat OpenShift Container Platform version 3.11 is deployed.
- At least four gigabytes of memory are available in the OpenShift cluster/namespace.
- The OpenShift project for the deployment has been created.
- You are logged in to the project using the **oc** command. For more information about the **oc** command-line tool, see the OpenShift CLI Reference. If you want to use the OpenShift Web console to deploy templates, you must also be logged on using the Web console.
- Dynamic persistent volume (PV) provisioning is enabled. Alternatively, if dynamic PV provisioning is not enabled, enough persistent volumes must be available. By default, the following sizes are required:
  - The replicated set of Process Server pods requires one 1Gi PV for the database by default. You can change the database PV size in the template parameters. This requirement does not apply if you use an external database server.
  - Business Central requires one 1Gi PV by default. You can change the PV size for Business Central persistent storage in the template parameters.
- Your OpenShift environment supports persistent volumes with **ReadWriteMany** mode. For information about access mode support in OpenShift Online volume plug-ins, see Access Modes.



IMPORTANT

ReadWriteMany mode is not supported on OpenShift Online and OpenShift Dedicated.

NOTE

Since Red Hat Process Automation Manager version 7.5, support for Red Hat OpenShift Container Platform 3.x is deprecated, including installation using all templates and using the Automation Broker (Ansible Playbook). New features might not be added, and this functionality will be removed in a future release.

CHAPTER 1. OVERVIEW OF RED HAT PROCESS AUTOMATION MANAGER ON RED HAT OPENSHIFT CONTAINER PLATFORM

You can deploy Red Hat Process Automation Manager into a Red Hat OpenShift Container Platform environment.

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

The following key components of Red Hat Process Automation Manager are available on OpenShift:

- Process Server, also known as *Execution Server* or *KIE Server*, is the infrastructure element that runs decision services, process applications, and other deployable assets (collectively referred to as *services*) . All logic of the services runs on execution servers.
  A database server is normally required for Process Server. You can provide a database server in another OpenShift pod or configure an execution server on OpenShift to use any other database server. Alternatively, Process Server can use an H2 database; in this case, you cannot scale the pod.

  You can scale up a Process Server pod to provide as many copies as required, running on the same host or different hosts. As you scale a pod up or down, all of its copies use the same database server and run the same services. OpenShift provides load balancing and a request can be handled by any of the pods.

  You can deploy a separate Process Server pod to run a different group of services. That pod can also be scaled up or down. You can have as many separate replicated Process Server pods as required.

- Business Central is a web-based interactive environment used for authoring services. It also provides a management and monitoring console. You can use Business Central to develop services and deploy them to Process Servers. You can also use Business Central to monitor the execution of processes.
  Business Central is a centralized application. However, you can configure it for high availability, where multiple pods run and share the same data.

  Business Central includes a Git repository that holds the source for the services that you develop on it. It also includes a built-in Maven repository. Depending on configuration, Business Central can place the compiled services (KJAR files) into the built-in Maven repository or (if configured) into an external Maven repository.

  IMPORTANT

  In the current version, high-availability Business Central functionality is for Technology Preview only. For more information on Red Hat Technology Preview features, see Technology Preview Features Scope .

- Business Central Monitoring is a web-based management and monitoring console. It can manage the deployment of services to Process Servers and provide monitoring information, but does not include authoring capabilities. You can use this component to manage staging and production environments.

- Smart Router is an optional layer between Process Servers and other components that interact with them. When your environment includes many services running on different Process

Servers, Smart Router provides a single endpoint to all client applications. A client application can make a REST API call that requires any service. Smart Router automatically calls the Process Server that can process a particular request.

You can arrange these and other components into various environment configurations within OpenShift.

The following environment types are typical:

- *Authoring*: An environment for creating and modifying services using Business Central. It consists of pods that provide Business Central for the authoring work and a Process Server for test execution of the services. For instructions about deploying this environment, see *Deploying a Red Hat Process Automation Manager authoring environment on Red Hat OpenShift Container Platform*.

- *Managed deployment*: An environment for running existing services for staging and production purposes. This environment includes several groups of Process Server pods; you can deploy and undeploy services on every such group and also scale the group up or down as necessary. Use Business Central Monitoring to deploy, run, and stop the services and to monitor their execution. You can deploy two types of managed environment. In a *freeform* server environment, you initially deploy Business Central Monitoring and one Process Server. You can additionally deploy any number of Process Servers. Business Central Monitoring can connects to all servers in the same namespace. For instructions about deploying this environment, see *Deploying a Red Hat Process Automation Manager freeform managed server environment on Red Hat OpenShift Container Platform*.

  Alternatively, you can deploy a *fixed* managed server environment. A single deployment includes Business Central Monitoring, Smart Router, and a preset number of Process Servers (by default, two servers, but you can modify the template to change the number). You cannot easily add or remove servers at a later time. For instructions about deploying this environment, see *Deploying a Red Hat Process Automation Manager fixed managed server environment on Red Hat OpenShift Container Platform*.

- *Deployment with immutable servers*: An alternate environment for running existing services for staging and production purposes. In this environment, when you deploy a Process Server pod, it builds an image that loads and starts a service or group of services. You cannot stop any service on the pod or add any new service to the pod. If you want to use another version of a service or modify the configuration in any other way, you deploy a new server image and displace the old one. In this system, the Process Server runs like any other pod on the OpenShift environment; you can use any container-based integration workflows and do not need to use any other tools to manage the pods. Optionally, you can use Business Central Monitoring to monitor the performance of the environment and to stop and restart some of the service instances, but not to deploy additional services to any Process Server or undeploy any existing ones (you cannot add or remove containers). For instructions about deploying this environment, see *Deploying a Red Hat Process Automation Manager immutable server environment on Red Hat OpenShift Container Platform*.

You can also deploy a *trial* or evaluation environment. This environment includes Business Central and a Process Server. You can set it up quickly and use it to evaluate or demonstrate developing and running assets. However, the environment does not use any persistent storage, and any work you do in the environment is not saved. For instructions about deploying this environment, see *Deploying a Red Hat Process Automation Manager trial environment on Red Hat OpenShift Container Platform*.

To deploy a Red Hat Process Automation Manager environment on OpenShift, you can use the templates that are provided with Red Hat Process Automation Manager. You can modify the templates to ensure that the configuration suits your environment.

CHAPTER 2. PREPARING TO DEPLOY RED HAT PROCESS AUTOMATION MANAGER IN YOUR OPENSHIFT ENVIRONMENT

Before deploying Red Hat Process Automation Manager in your OpenShift environment, you must complete several tasks. You do not need to repeat these tasks if you want to deploy additional images, for example, for new versions of processes or for other processes.

2.1. ENSURING THE AVAILABILITY OF IMAGE STREAMS AND THE IMAGE REGISTRY

To deploy Red Hat Process Automation Manager components on Red Hat OpenShift Container Platform, you must ensure that OpenShift can download the correct images from the Red Hat registry. To download the images, OpenShift requires *image streams*, which contain the information about the location of images. OpenShift also must be configured to authenticate with the Red Hat registry using your service account user name and password.

Some versions of the OpenShift environment include the required image streams. You must check if they are available. If image streams are available in OpenShift by default, you can use them if the OpenShift infrastructure is configured for registry authentication server. The administrator must complete the registry authentication configuration when installing the OpenShift environment.

Otherwise, you can configure registry authentication in your own project and install the image streams in that project.

Procedure

1. Determine whether Red Hat OpenShift Container Platform is configured with the user name and password for Red Hat registry access. For details about the required configuration, see Configuring a Registry Location. If you are using an OpenShift Online subscription, it is configured for Red Hat registry access.

2. If Red Hat OpenShift Container Platform is configured with the user name and password for Red Hat registry access, enter the following commands:

```
$ oc get imagestreamtag -n openshift | grep -F rhpam-businesscentral | grep -F 7.5
$ oc get imagestreamtag -n openshift | grep -F rhpam-kieserver | grep -F 7.5
```

   If the outputs of both commands are not empty, the required image streams are available in the **openshift** namespace and no further action is required.

3. If the output of one or both of the commands is empty or if OpenShift is not configured with the user name and password for Red Hat registry access, complete the following steps:

   a. Ensure you are logged in to OpenShift with the **oc** command and that your project is active.

   b. Complete the steps documented in Registry Service Accounts for Shared Environments. You must log in to the Red Hat Customer Portal to access the document and to complete the steps to create a registry service account.

   c. Select the **OpenShift Secret** tab and click the link under **Download secret** to download the YAML secret file.

   d. View the downloaded file and note the name that is listed in the **name:** entry.

e. Enter the following commands:

```
oc create -f <file_name>.yaml
oc secrets link default <secret_name> --for=pull
oc secrets link builder <secret_name> --for=pull
```

Replace **<file_name>** with the name of the downloaded file and **<secret_name>** with the name that is listed in the **name:** entry of the file.

f. Download the **rhpam-7.5.1-openshift-templates.zip** product deliverable file from the Software Downloads page and extract the **rhpam75-image-streams.yaml** file.

g. Enter the following command:

```
$ oc apply -f rhpam75-image-streams.yaml
```

NOTE

If you complete these steps, you install the image streams into the namespace of your project. In this case, when you deploy the templates, you must set the **IMAGE_STREAM_NAMESPACE** parameter to the name of this project.

2.2. CREATING THE SECRETS FOR PROCESS SERVER

OpenShift uses objects called *secrets* to hold sensitive information such as passwords or keystores. For more information about OpenShift secrets, see the Secrets chapter in the OpenShift documentation.

You must create an SSL certificate for HTTP access to Process Server and provide it to your OpenShift environment as a secret.

Procedure

1. Generate an SSL keystore with a private and public key for SSL encryption for Process Server. For more information on how to create a keystore with self-signed or purchased SSL certificates, see Generate a SSL Encryption Key and Certificate .

NOTE

In a production environment, generate a valid signed certificate that matches the expected URL for Process Server.

2. Save the keystore in a file named **keystore.jks**.

3. Record the name of the certificate. The default value for this name in Red Hat Process Automation Manager configuration is **jboss**.

4. Record the password of the keystore file. The default value for this name in Red Hat Process Automation Manager configuration is **mykeystorepass**.

5. Use the **oc** command to generate a secret named **kieserver-app-secret** from the new keystore file:

```
$ oc create secret generic kieserver-app-secret --from-file=keystore.jks
```

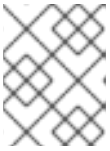
2.3. CREATING THE SECRETS FOR BUSINESS CENTRAL

You must create an SSL certificate for HTTP access to Business Central and provide it to your OpenShift environment as a secret.

Do not use the same certificate and keystore for Business Central and Process Server.

Procedure

1. Generate an SSL keystore with a private and public key for SSL encryption for Business Central. For more information on how to create a keystore with self-signed or purchased SSL certificates, see Generate a SSL Encryption Key and Certificate .



NOTE

In a production environment, generate a valid signed certificate that matches the expected URL for Business Central.

2. Save the keystore in a file named **keystore.jks**.
3. Record the name of the certificate. The default value for this name in Red Hat Process Automation Manager configuration is **jboss**.
4. Record the password of the keystore file. The default value for this name in Red Hat Process Automation Manager configuration is **mykeystorepass**.
5. Use the **oc** command to generate a secret named **businesscentral-app-secret** from the new keystore file:

```
$ oc create secret generic businesscentral-app-secret --from-file=keystore.jks
```

2.4. CHANGING GLUSTERFS CONFIGURATION

You must check whether your OpenShift environment uses GlusterFS to provide permanent storage volumes. If it uses GlusterFS, to ensure optimal performance of Business Central, you must tune your GlusterFS storage by changing the storage class configuration.

Procedure

1. To check whether your environment uses GlusterFS, enter the following command:

```
oc get storageclass
```

In the results, check whether the **(default)** marker is on the storage class that lists **glusterfs**. For example, in the following output the default storage class is **gluster-container**, which does list **glusterfs**:

```
NAME              PROVISIONER                AGE
gluster-block     gluster.org/glusterblock   8d
gluster-container (default) kubernetes.io/glusterfs 8d
```

If the result has a default storage class that does not list **glusterfs** or if the result is empty, you do not need to make any changes. In this case, skip the rest of this procedure.

2. To save the configuration of the default storage class into a YAML file, enter the following command:

```
oc get storageclass <class-name> -o yaml >storage_config.yaml
```

Replace **<class-name>** with the name of the default storage class. Example:

```
oc get storageclass gluster-container -o yaml >storage_config.yaml
```

3. Edit the **storage_config.yaml** file:

a. Remove the lines with the following keys:

- **creationTimestamp**
- **resourceVersion**
- **selfLink**
- **uid**

b. If you are planning to use Business Central only as a single pod, without high-availability configuration, on the line with the **volumeoptions** key, add the following options:

```
features.cache-invalidation on
performance.nl-cache on
```

For example:

volumeoptions: client.ssl off, server.ssl off, features.cache-invalidation on, performance.nl-cache on

c. If you are planning to use Business Central in a high-availability configuration, on the line with the **volumeoptions** key, add the following options:

```
features.cache-invalidation on
nfs.trusted-write on
nfs.trusted-sync on
performance.nl-cache on
performance.stat-prefetch off
performance.read-ahead off
performance.write-behind off
performance.readdir-ahead off
performance.io-cache off
performance.quick-read off
performance.open-behind off
locks.mandatory-locking off
performance.strict-o-direct on
```

For example:

volumeoptions: client.ssl off, server.ssl off, features.cache-invalidation on, nfs.trusted-write on, nfs.trusted-sync on, performance.nl-cache on, performance.stat-

prefetch off, performance.read-ahead off, performance.write-behind off, performance.readdir-ahead off, performance.io-cache off, performance.quick-read off, performance.open-behind off, locks.mandatory-locking off, performance.strict-o-direct on

4. To remove the existing default storage class, enter the following command:

```
oc delete storageclass <class-name>
```

Replace **<class-name>** with the name of the default storage class. Example:

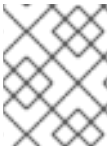
```
oc delete storageclass gluster-container
```

5. To re-create the storage class using the new configuration, enter the following command:

```
oc create -f storage_config.yaml
```

2.5. PREPARING A MAVEN MIRROR REPOSITORY FOR OFFLINE USE

If your Red Hat OpenShift Container Platform environment 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 OpenShift Container Platform environment is connected to the Internet.

Prerequisites

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

Procedure

1. Prepare a Maven release repository to which you can write. The repository must allow read access without authentication. Your OpenShift environment must have access to this repository. You can deploy a Nexus repository manager in the OpenShift environment. For instructions about setting up Nexus on OpenShift, see Setting up Nexus. Use this repository as a separate mirror repository.
   Alternatively, if you use a custom external repository (for example, Nexus) for your services, you can use the same repository as a mirror repository.

2. On the computer that has an outgoing connection to the public Internet, complete the following steps:

   a. Download the latest version of the Offliner tool.

   b. Download the **rhpam-7.5.1-offliner.txt** product deliverable file from the Software Downloads page of the Red Hat Customer Portal.

   c. Enter the following command to use the Offliner tool to download the required artifacts:

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

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

d. Upload all artifacts from the temporary directory to the Maven mirror repository that you prepared. You can use the Maven Repository Provisioner utility to upload the artifacts.

3. If you developed services outside Business Central and they have additional dependencies, add the dependencies to the mirror repository. If you developed the services as Maven projects, you can use the following steps to prepare these dependencies automatically. Complete the steps on the computer that has an outgoing connection to the public Internet.

a. Create a backup of the local Maven cache directory (**~/.m2/repository**) and then clear the directory.

b. Build the source of your projects using the **mvn clean install** command.

c. For every project, enter the following command to ensure that Maven downloads all runtime dependencies for all the artifacts generated by the project:

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

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

d. Upload all artifacts from the local Maven cache directory (**~/.m2/repository**) to the Maven mirror repository that you prepared. You can use the Maven Repository Provisioner utility to upload the artifacts.

2.6. BUILDING A CUSTOM PROCESS SERVER EXTENSION IMAGE FOR AN EXTERNAL DATABASE

If you want to use an external database server for a Process Server and the database server is not a MySQL or PostgreSQL server, you must build a custom Process Server extension image with drivers for this server before deploying your environment.

Complete the steps in this build procedure to provide drivers for any of the following database servers:

- Microsoft SQL Server
- MariaDB
- IBM DB2
- Oracle Database
- Sybase

For the supported versions of the database servers, see Red Hat Process Automation Manager 7 Supported Configurations.

The build procedure creates a custom extension image that extends the existing Process Server image. You must import this custom extension image into your OpenShift environment and then reference it in the **EXTENSION_IMAGE** parameter.

Prerequisites

- You are logged in to your OpenShift environment using the **oc** command. Your OpenShift user must have the **registry-editor** role.
- For Oracle Database or Sybase, you downloaded the JDBC driver from the database server vendor.
- You have installed the following required software:
  - Docker
  - Cekit version 3.2
  - The following libraries and extensions for Cekit:
    - **odcs-client**, provided by the **python3-odcs-client** package or similar package
    - **docker**, provided by the **python3-docker** package or similar package
    - **docker-squash**, provided by the **python3-docker-squash** package or similar package
    - **behave**, provided by the **python3-behave** package or similar package
    - **s2i**, provided by the **source-to-image** package or similar package

Procedure

1. For IBM DB2, Oracle Database, or Sybase, provide the JDBC driver JAR file in a local directory.
2. Download the **rhpam-7.5.1-openshift-templates.zip** product deliverable file from the Software Downloads page of the Red Hat Customer Portal.
3. Unzip the file and, using the command line, change to the **templates/contrib/jdbc** directory of the unzipped file. This directory contains the source code for the custom build.
4. Run one of the following commands, depending on the database server type:
   - For Microsoft SQL Server:

     ```
     make build mssql
     ```

   - For MariaDB:

     ```
     make build mariadb
     ```

   - For IBM DB2:

     ```
     make build db2
     ```

   - For Oracle Database:

     ```
     make build oracle artifact=/tmp/ojdbc7.jar version=7.0
     ```

     In this command, replace **/tmp/ojdbc7.jar** with the path name of the downloaded Oracle Database driver and **7.0** with the version of the driver.

- For Sybase:

```
make build sybase artifact=/tmp/jconn4-16.0_PL05.jar version=16.0_PL05
```

In this command, replace **/tmp/jconn4-16.0_PL05.jar** with the path name of the downloaded Sybase driver and **16.0_PL05** with the version of the driver.

5. Run the following command to list the Docker images that are available locally:

```
docker images
```

Note the name of the image that was built, for example, **jboss-kie-db2-extension-openshift-image**, and the version tag of the image, for example, **11.1.4.4** (not the **latest** tag).

6. Access the registry of your OpenShift environment directly and push the image to the registry. Depending on your user permissions, you can push the image into the **openshift** namespace or into a project namespace. For instructions about accessing the registry and pushing the images, see Accessing the Registry Directly.

7. When configuring your Process Server deployment with a template that supports an external database server, set the following parameters:

- **Drivers Extension Image** (**EXTENSIONS_IMAGE**): The ImageStreamTag definition of the extension image, for example, **jboss-kie-db2-extension-openshift-image:11.1.4.4**
- **Drivers ImageStream Namespace** (**EXTENSIONS_IMAGE_NAMESPACE**): The namespace to which you uploaded the extension image, for example, **openshift** or your project namespace.

CHAPTER 3. AUTHORING ENVIRONMENT

You can deploy an environment for creating and modifying processes using Business Central. It consists of Business Central for the authoring work and Process Server for test execution of the processes.

Depending on your needs, you can deploy either a single authoring environment or a high-availability (HA) authoring environment.

A single authoring environment contains two pods. One of the pods runs Business Central, the other runs Process Server. The Process Server includes an embedded in-memory H2 database engine. This type of environment uses the least possible amount of resources. However, because of the in-memory database, restarting the Process Server pod leads to loss of all process information.

An HA authoring environment contains several pods. Both Business Central and Process Server are provided in scalable pods that can run in parallel and share persistent storage. The database is provided by a separate pod. Use a high-availability authoring environment to provide maximum reliability and responsiveness, especially if several users are involved in authoring at the same time.

You can also deploy additional managed or immutable Process Servers, if required. Business Central can automatically discover any Process Servers in the same namespace, including immutable Process Servers and managed Process Servers. This feature requires the **OpenShiftStartupStrategy** setting, which is enabled for all Process Servers except those deployed in a fixed managed infrastructure. For instructions about deploying managed Process Servers with the **OpenShiftStartupStrategy** setting enabled, see *Deploying a Red Hat Process Automation Manager freeform managed server environment on Red Hat OpenShift Container Platform*. For instructions about deploying immutable Process Servers, see *Deploying a Red Hat Process Automation Manager immutable server environment on Red Hat OpenShift Container Platform*.

IMPORTANT

In Red Hat Process Automation Manager 7.5, high-availability Business Central functionality is for Technology Preview only. For more information on Red Hat Technology Preview features, see Technology Preview Features Scope .

3.1. DEPLOYING AN AUTHORING ENVIRONMENT

You can use OpenShift templates to deploy a single or high-availability authoring environment. This environment consists of Business Central and a single Process Server.

3.1.1. Starting configuration of the template for an authoring environment

If you want to deploy a single authoring environment, use the **rhpam75-authoring.yaml** template file. By default, the single authoring template uses the H2 database with permanent storage. If you prefer to create a MySQL or PostgreSQL pod or to use an external database server (outside the OpenShift project), modify the template before deploying the environment. For instructions about modifying the template, see Section 3.4, “Modifying the template for the single authoring environment”.

If you want to deploy a high-availability authoring environment, use the **rhpam75-authoring-ha.yaml** template file. By default, the high-availability authoring template creates a MySQL pod to provide the database server for the Process Server. If you prefer to use PostgreSQL or to use an external server (outside the OpenShift project) you need to modify the template before deploying the environment. You can also modify the template to change the number of replicas initially created for Business Central. For instructions about modifying the template, see Section 3.5, “Modifying the template for the High Availability authoring environment”.

Procedure

1. Download the **rhpam-7.5.1-openshift-templates.zip** product deliverable file from the Software Downloads page of the Red Hat Customer Portal.
2. Extract the required template file.
3. Use one of the following methods to start deploying the template:
   - To use the OpenShift Web UI, in the OpenShift application console select **Add to Project → Import YAML / JSON** and then select or paste the **<template-file-name>.yaml** file. In the **Add Template** window, ensure **Process the template** is selected and click **Continue**.
   - To use the OpenShift command line console, prepare the following command line:

     ```
     oc new-app -f <template-path>/<template-file-name>.yaml -p
     BUSINESS_CENTRAL_HTTPS_SECRET=businesscentral-app-secret -p
     KIE_SERVER_HTTPS_SECRET=kieserver-app-secret -p PARAMETER=value
     ```

     In this command line, make the following changes:
     - Replace **<template-path>** with the path to the downloaded template file.
     - Replace **<template-file-name>** with the name of the template file.
     - Use as many **-p PARAMETER=value** pairs as needed to set the required parameters.

Next steps

Set the parameters for the template. Follow the steps in Section 3.1.2, "Setting required parameters for an authoring environment" to set common parameters. You can view the template file to see descriptions for all parameters.

3.1.2. Setting required parameters for an authoring environment

When configuring the template to deploy an authoring environment, you must set the following parameters in all cases.

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

1. Set the following parameters:
   - **Business Central Server Keystore Secret Name** (**BUSINESS_CENTRAL_HTTPS_SECRET**): The name of the secret for Business Central, as created in Section 2.3, "Creating the secrets for Business Central" .
   - **KIE Server Keystore Secret Name**(**KIE_SERVER_HTTPS_SECRET**): The name of the secret for Process Server, as created in Section 2.2, "Creating the secrets for Process Server".

- **Business Central Server Certificate Name**(**BUSINESS_CENTRAL_HTTPS_NAME**): The name of the certificate in the keystore that you created in Section 2.3, "Creating the secrets for Business Central".
- **Business Central Server Keystore Password** (**BUSINESS_CENTRAL_HTTPS_PASSWORD**): The password for the keystore that you created in Section 2.3, "Creating the secrets for Business Central" .
- **KIE Server Certificate Name**(**KIE_SERVER_HTTPS_NAME**): The name of the certificate in the keystore that you created in Section 2.2, "Creating the secrets for Process Server" .
- **KIE Server Keystore Password** (**KIE_SERVER_HTTPS_PASSWORD**): The password for the keystore that you created in Section 2.2, "Creating the secrets for Process Server" .
- **Application Name** (**APPLICATION_NAME**): The name of the OpenShift application. It is used in the default URLs for Business Central Monitoring and Process Server. OpenShift uses the application name to create a separate set of deployment configurations, services, routes, labels, and artifacts.
- **Enable KIE server global discovery** (**KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED**): Set this parameter to **true** if you want Business Central to discover all Process Servers with the **OpenShiftStartupStrategy** in the same namespace. By default, Business Central discovers only Process Servers that are deployed with the same value of the **APPLICATION_NAME** parameter as Business Central itself.
- **ImageStream Namespace** (**IMAGE_STREAM_NAMESPACE**): The namespace where the image streams are available. If the image streams were already available in your OpenShift environment (see Section 2.1, "Ensuring the availability of image streams and the image registry"), the namespace is **openshift**. If you have installed the image streams file, the namespace is the name of the OpenShift project.

2. You can set the following user names and passwords. By default, the deployment automatically generates the passwords.
   - **KIE Admin User** (**KIE_ADMIN_USER**) and **KIE Admin Password** (**KIE_ADMIN_PWD**): The user name and password for the administrative user. If you want to use the Business Central to control or monitor any Process Servers other than the Process Server deployed by the same template , you must set and record the user name and password.
   - **KIE Server User** (**KIE_SERVER_USER**) and **KIE Server Password** (**KIE_SERVER_PWD**): The user name and password that a client application can use to connect to any of the Process Servers.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

3.1.3. Configuring the image stream namespace for an authoring environment

If you created image streams in a namespace that is not **openshift**, you must configure the namespace in the template.

If all image streams were already available in your Red Hat OpenShift Container Platform environment, you can skip this procedure.

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

If you installed an image streams file according to instructions in Section 2.1, "Ensuring the availability of image streams and the image registry", set the **ImageStream Namespace** (**IMAGE_STREAM_NAMESPACE**) parameter to the name of your OpenShift project.

3.1.4. Setting an optional Maven repository for an authoring environment

When configuring the template to deploy an authoring environment, if you want to place the built KJAR files into an external Maven repository, you must set parameters to access the repository.

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

To configure access to a custom Maven repository, set the following parameters:

- **Maven repository URL** (**MAVEN_REPO_URL**): The URL for the Maven repository.
- **Maven repository ID** (**MAVEN_REPO_ID**): An identifier for the Maven repository. The default value is **repo-custom**.
- **Maven repository username** (**MAVEN_REPO_USERNAME**): The username for the Maven repository.
- **Maven repository password** (**MAVEN_REPO_PASSWORD**): The password for the Maven repository.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

IMPORTANT

To export or push Business Central projects as KJAR artifacts to the external Maven repository, you must also add the repository information in the **pom.xml** file for every project. For information about exporting Business Central projects to an external repository, see *Packaging and deploying a Red Hat Process Automation Manager project* .

3.1.5. Specifying credentials to access the built-in Maven repository for an authoring environment

When configuring the template to deploy an authoring environment, if you want to use the Maven repository that is built into Business Central and to connect additional Process Servers to the Business Central, you must configure credentials for accessing this Maven repository. You can then use these credentials to configure the Process Servers.

Also, if you are configuring RH-SSO or LDAP authentication, you must set the credentials for the built-in Maven repository to a username and password configured in RH-SSO or LDAP. This setting is required so that the Process Server can access the Maven repository.

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

To configure credentials for the built-in Maven repository, set the following parameters:

- **Username for the Maven service hosted by Business Central** (**BUSINESS_CENTRAL_MAVEN_USERNAME**): The user name for the built-in Maven repository.
- **Password for the Maven service hosted by Business Central** (**BUSINESS_CENTRAL_MAVEN_PASSWORD**): The password for the built-in Maven repository.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

3.1.6. Configuring access to a Maven mirror in an environment without a connection to the public Internet for an authoring environment

When configuring the template to deploy an authoring environment, if your OpenShift environment does not have a connection to the public Internet, you must configure access to a Maven mirror that you set up according to Section 2.5, "Preparing a Maven mirror repository for offline use" .

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

To configure access to the Maven mirror, set the following parameters:

- **Maven mirror URL** (**MAVEN_MIRROR_URL**): The URL for the Maven mirror repository that you set up in Section 2.5, "Preparing a Maven mirror repository for offline use" . This URL must be accessible from a pod in your OpenShift environment.
- **Maven mirror of** (**MAVEN_MIRROR_OF**): The value that determines which artifacts are to be retrieved from the mirror. For instructions about setting the **mirrorOf** value, see Mirror Settings in the Apache Maven documentation. The default value is **external:*,!repo-rhpamcentr**; with this value, Maven retrieves artifacts from the built-in Maven repository of Business Central

directly and retrieves any other required artifacts from the mirror. If you configure an external Maven repository (**MAVEN_REPO_URL**), change **MAVEN_MIRROR_OF** to exclude the artifacts in this repository, for example, **external:*,!repo-custom**. Replace **repo-custom** with the ID that you configured in **MAVEN_REPO_ID**.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

3.1.7. Specifying the Git hooks directory for an authoring environment

You can use Git hooks to facilitate interaction between the internal Git repository of Business Central and an external Git repository.

If you want to use Git hooks, you must configure a Git hooks directory.

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

To configure a Git hooks directory, set the following parameter:

- **Git hooks directory** (**GIT_HOOKS_DIR**): The fully qualified path to a Git hooks directory, for example, **/opt/kie/data/git/hooks**. You must provide the content of this directory and mount it at the specified path. For instructions about providing and mounting the Git hooks directory using a configuration map or a persistent volume, see Section 3.3, "(Optional) Providing the Git hooks directory".

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

3.1.8. Setting parameters for RH-SSO authentication for an authoring environment

If you want to use RH-SSO authentication, complete the following additional configuration when configuring the template to deploy an authoring environment.

IMPORTANT

Do not configure LDAP authentication and RH-SSO authentication in the same deployment.

Prerequisites

- A realm for Red Hat Process Automation Manager is created in the RH-SSO authentication system.

- User names and passwords for Red Hat Process Automation Manager are created in the RH-SSO authentication system. For a list of the available roles, see Chapter 4, *Red Hat Process Automation Manager roles and users*. The following users are required in order to set the parameters for the environment:
  - An administrative user with the **kie-server,rest-all,admin** roles. This user can administer and use the environment. Process Servers use this user to authenticate with Business Central.
  - A server user with the **kie-server,rest-all,user** roles. This user can make REST API calls to the Process Server. Business Central uses this user to authenticate with Process Servers.
- Clients are created in the RH-SSO authentication system for all components of the Red Hat Process Automation Manager environment that you are deploying. The client setup contains the URLs for the components. You can review and edit the URLs after deploying the environment. Alternatively, the Red Hat Process Automation Manager deployment can create the clients. However, this option provides less detailed control over the environment.
- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

1. Set the **KIE_ADMIN_USER** and **KIE_ADMIN_PASSWORD** parameters of the template to the user name and password of the administrative user that you created in the RH-SSO authentication system.
2. Set the **KIE_SERVER_USER** and **KIE_SERVER_PASSWORD** parameters of the template to the user name and password of the server user that you created in the RH-SSO authentication system.
3. Set the following parameters:
   - **RH-SSO URL** (**SSO_URL**): The URL for RH-SSO.
   - **RH-SSO Realm name** (**SSO_REALM**): The RH-SSO realm for Red Hat Process Automation Manager.
   - **RH-SSO Disable SSL Certificate Validation** (**SSO_DISABLE_SSL_CERTIFICATE_VALIDATION**): Set to **true** if your RH-SSO installation does not use a valid HTTPS certificate.
4. Complete one of the following procedures:
   a. If you created the clients for Red Hat Process Automation Manager within RH-SSO, set the following parameters in the template:
      - **Business Central RH-SSO Client name** (**BUSINESS_CENTRAL_SSO_CLIENT**): The RH-SSO client name for Business Central.
      - **Business Central RH-SSO Client Secret** (**BUSINESS_CENTRAL_SSO_SECRET**): The secret string that is set in RH-SSO for the client for Business Central.
      - **KIE Server RH-SSO Client name** (**KIE_SERVER_SSO_CLIENT**): The RH-SSO client name for Process Server.
      - **KIE Server RH-SSO Client Secret** (**KIE_SERVER_SSO_SECRET**): The secret string that is set in RH-SSO for the client for Process Server.

b. To create the clients for Red Hat Process Automation Manager within RH-SSO, set the following parameters in the template:

- **Business Central RH-SSO Client name** (**BUSINESS_CENTRAL_SSO_CLIENT**): The name of the client to create in RH-SSO for Business Central.
- **Business Central RH-SSO Client Secret** (**BUSINESS_CENTRAL_SSO_SECRET**): The secret string to set in RH-SSO for the client for Business Central.
- **KIE Server RH-SSO Client name** (**KIE_SERVER_SSO_CLIENT**): The name of the client to create in RH-SSO for Process Server.
- **KIE Server RH-SSO Client Secret** (**KIE_SERVER_SSO_SECRET**): The secret string to set in RH-SSO for the client for Process Server.
- **RH-SSO Realm Admin Username** (**SSO_USERNAME**) and **RH-SSO Realm Admin Password** (**SSO_PASSWORD**): The user name and password for the realm administrator user for the RH-SSO realm for Red Hat Process Automation Manager. You must provide this user name and password in order to create the required clients.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

After completing the deployment, review the URLs for components of Red Hat Process Automation Manager in the RH-SSO authentication system to ensure they are correct.

3.1.9. Setting parameters for LDAP authentication for an authoring environment

If you want to use LDAP authentication, complete the following additional configuration when configuring the template to deploy an authoring environment.

IMPORTANT

Do not configure LDAP authentication and RH-SSO authentication in the same deployment.

Prerequisites

- You created user names and passwords for Red Hat Process Automation Manager in the LDAP system. For a list of the available roles, see Chapter 4, *Red Hat Process Automation Manager roles and users*. As a minimum, in order to set the parameters for the environment, you created the following users:
  - An administrative user with the **kie-server,rest-all,admin** roles. This user can administer and use the environment.
  - A server user with the **kie-server,rest-all,user** roles. This user can make REST API calls to the Process Server.
- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

1. In the LDAP service, create all user names in the deployment parameters. If you do not set any of the parameters, create users with the default user names. The created users must also be assigned to roles:
   - **KIE_ADMIN_USER**: default user name **adminUser**, roles: **kie-server,rest-all,admin**
   - **KIE_SERVER_USER**: default user name **executionUser**, roles **kie-server,rest-all,guest**
     For the user roles that you can configure in LDAP, see Roles and users.
2. Set the **AUTH_LDAP*** parameters of the template. These parameters correspond to the settings of the **LdapExtended** Login module of Red Hat JBoss EAP. For instructions about using these settings, see LdapExtended login module .
   If the LDAP server does not define all the roles required for your deployment, you can map LDAP groups to Red Hat Process Automation Manager roles. To enable LDAP role mapping, set the following parameters:
   - **RoleMapping rolesProperties file path** (**AUTH_ROLE_MAPPER_ROLES_PROPERTIES**): The fully qualified path name of a file that defines role mapping, for example, **/opt/eap/standalone/configuration/rolemapping/rolemapping.properties**. You must provide this file and mount it at this path in all applicable deployment configurations; for instructions, see Section 3.2, "(Optional) Providing the LDAP role mapping file" .
   - **RoleMapping replaceRole property** (**AUTH_ROLE_MAPPER_REPLACE_ROLE**): If set to **true**, mapped roles replace the roles defined on the LDAP server; if set to **false**, both mapped roles and roles defined on the LDAP server are set as user application roles. The default setting is **false**.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

3.1.10. Setting parameters for using an external database server for an authoring environment

If you modified the template to use an external database server for the Process Server, as described in Section 3.4, "Modifying the template for the single authoring environment" or Section 3.5, "Modifying the template for the High Availability authoring environment", complete the following additional configuration when configuring the template to deploy an authoring environment.

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

1. Set the following parameters:
   - **KIE Server External Database Driver**(**KIE_SERVER_EXTERNALDB_DRIVER**): The driver for the server, depending on the server type:

- **mysql**
- **postgresql**
- **mariadb**
- **mssql**
- **db2**
- **oracle**
- **sybase**

- KIE Server External Database User(**KIE_SERVER_EXTERNALDB_USER**) and KIE Server External Database Password (**KIE_SERVER_EXTERNALDB_PWD**): The user name and password for the external database server
- KIE Server External Database URL(**KIE_SERVER_EXTERNALDB_URL**): The JDBC URL for the external database server
- KIE Server External Database Dialect(**KIE_SERVER_EXTERNALDB_DIALECT**): The Hibernate dialect for the server, depending on the server type:
  - **org.hibernate.dialect.MySQL5InnoDBDialect** (used for MySQL and MariaDB)
  - **org.hibernate.dialect.PostgreSQL82Dialect**
  - **org.hibernate.dialect.SQLServer2012Dialect** (used for MS SQL)
  - **org.hibernate.dialect.DB2Dialect**
  - **org.hibernate.dialect.Oracle10gDialect**
  - **org.hibernate.dialect.SybaseASE157Dialect**
- KIE Server External Database Host(**KIE_SERVER_EXTERNALDB_SERVICE_HOST**): The host name of the external database server
- KIE Server External Database Port(**KIE_SERVER_EXTERNALDB_SERVICE_PORT**): The port number of the external database server
- KIE Server External Database name(**KIE_SERVER_EXTERNALDB_DB**): The database name to use on the external database server
- JDBC Connection Checker class (**KIE_SERVER_EXTERNALDB_CONNECTION_CHECKER**): The name of the JDBC connection checker class for the database server. Without this information, a database server connection cannot be restored after it is lost, for example, if the database server is rebooted.
- JDBC Exception Sorter class (**KIE_SERVER_EXTERNALDB_EXCEPTION_SORTER**): The name of the JDBC exception sorter class for the database server. Without this information, a database server connection cannot be restored after it is lost, for example, if the database server is rebooted.

2. If you created a custom image for using an external database server other than MySQL or PostgreSQL, as described in Section 2.6, "Building a custom Process Server extension image for an external database", set the following parameters:

    - **Drivers Extension Image** (**EXTENSIONS_IMAGE**): The ImageStreamTag definition of the extension image, for example, **jboss-kie-db2-extension-openshift-image:11.1.4.4**
    - **Drivers ImageStream Namespace** (**EXTENSIONS_IMAGE_NAMESPACE**): The namespace to which you uploaded the extension image, for example, **openshift** or your project namespace.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

3.1.11. Enabling Prometheus metric collection for an authoring environment

If you want to configure your Process Server deployment to use Prometheus to collect and store metrics, enable support for this feature in Process Server at deployment time.

Prerequisites

- You started the configuration of the template, as described in Section 3.1.1, "Starting configuration of the template for an authoring environment".

Procedure

To enable support for Prometheus metric collection, set the **Prometheus Server Extension Disabled** (**PROMETHEUS_SERVER_EXT_DISABLED**) parameter to **false**.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 3.1.12, "Completing deployment of the template for an authoring environment".

For instructions about configuring Prometheus metrics collection, see *Managing and monitoring Process Server*.

3.1.12. Completing deployment of the template for an authoring environment

After setting all the required parameters in the OpenShift Web UI or in the command line, complete deployment of the template.

Procedure

Depending on the method that you are using, complete the following steps:

- In the OpenShift Web UI, click **Create**.
    - If the **This will create resources that may have security or project behavior implications** message appears, click **Create Anyway**.
- Complete the command line and press Enter.

3.2. (OPTIONAL) PROVIDING THE LDAP ROLE MAPPING FILE

If you configure the **AUTH_ROLE_MAPPER_ROLES_PROPERTIES** parameter, you must provide a file that defines the role mapping. Mount this file on all affected deployment configurations.

Procedure

1. Create the role mapping properties file, for example, **my-role-map**. The file must contain entries in the following format:

```
ldap_role = product_role1, product_role2...
```

   For example:

```
admins = kie-server,rest-all,admin
```

2. Create an OpenShift configuration map from the file by entering the following command:

```
oc create configmap ldap-role-mapping --from-file=<new_name>=<existing_name>
```

   Replace **<new_name>** with the name that the file is to have on the pods (it must be the same as the name specified in the **AUTH_ROLE_MAPPER_ROLES_PROPERTIES** file) and **<existing_name>** with the name of the file that you created. Example:

```
oc create configmap ldap-role-mapping --from-file=rolemapping.properties=my-role-map
```

3. Mount the configuration map on every deployment configuration that is configured for role mapping.
   The following deployment configurations can be affected in this environment:

   - ***myapp*-rhpamcentr**: Business Central
   - ***myapp*-kieserver**: Process Server

   Replace **myapp** with the application name. Sometimes, several Process Server deployments can be present under different application names.

   For every deployment configuration, run the command:

```
 oc set volume dc/<deployment_config_name> --add --type configmap --configmap-name
ldap-role-mapping --mount-path=<mapping_dir> --name=ldap-role-mapping
```

   Replace **<mapping_dir>** with the directory name (without file name) set in the **AUTH_ROLE_MAPPER_ROLES_PROPERTIES** parameter, for example, **/opt/eap/standalone/configuration/rolemapping** .

3.3. (OPTIONAL) PROVIDING THE GIT HOOKS DIRECTORY

If you configure the **GIT_HOOKS_DIR** parameter, you must provide a directory of Git hooks and must mount this directory on the Business Central deployment.

The typical use of Git hooks is interaction with an upstream repository. To enable Git hooks to push commits into an upstream repository, you must also provide a secret key that corresponds to a public key configured on the upstream repository.

Procedure

1. If interaction with an upstream repository using SSH authentication is required, complete the following steps to prepare and mount a secret with the necessary files:
   a. Prepare the **id_rsa** file with a private key that matches a public key stored in the repository.
   b. Prepare the **known_hosts** file with the correct name, address, and public key for the repository.
   c. Create a secret with the two files using the **oc** command, for example:

```
oc create secret git-hooks-secret --from-file=id_rsa=id_rsa --from-file=known_hosts=known_hosts
```

   d. Mount the secret in the SSH key path of the Business Central deployment, for example:

```
oc set volume dc/<myapp>-rhpamcentr --add --type secret --secret-name git-hooks-secret --mount-path=/home/jboss/.ssh --name=ssh-key
```

   Replace **<myapp>** with the application name that you set when configuring the template.

2. Create the Git hooks directory. For instructions, see the Git hooks reference documentation. For example, a simple Git hooks directory can provide a post-commit hook that pushes the changes upstream. If the project was imported into Business Central from a repository, this repository remains configured as the upstream repository. Create a file named **post-commit** with permission values **755** and the following content:

```
git push
```

3. Supply the Git hooks directory to the Business Central deployment. You can use a configuration map or a persistent volume.
   a. If the Git hooks consist of one or several fixed script files, use a configuration map. Complete the following steps:
      i. Change into the Git hooks directory that you have created.
      ii. Create an OpenShift configuration map from the files in the directory. Run the following command:

```
oc create configmap git-hooks --from-file=<file_1>=<file_1> --from-file=<file_2>=<file_2> ...
```

      Replace **file_1**, **file_2**, and so on with Git hook script file names. Example:

```
oc create configmap git-hooks --from-file=post-commit=post-commit
```

      iii. Mount the configuration map on the Business Central deployment in the path that you have configured:

```
oc set volume dc/<myapp>-rhpamcentr --add --type configmap --configmap-name
git-hooks  --mount-path=<git_hooks_dir> --name=git-hooks
```

Replace **<myapp>** with the application name that was set when configuring the template and **<git_hooks_dir>** is the value of **GIT_HOOKS_DIR** that was set when configuring the template.

b. If the Git hooks consist of long files or depend on binaries, such as executable or KJAR files, use a persistence volume. You must create a persistent volume, create a persistent volume claim and associate the volume with the claim, transfer files to the volume, and mount the volume in the ***myapp*-rhpamcentr** deployment configuration (replace *myapp* with the application name). For instructions about creating and mounting persistence volumes, see Using persistent volumes. For instructions about copying files onto a persistent volume, see Transferring files in and out of containers .

4. Wait a few minutes, then review the list and status of pods in your project. Because Business Central does not start until you provide the Git hooks directory, the Process Server might not start at all. To see if it has started, check the output of the following command:

```
oc get pods
```

If a working Process Server pod is not present, start it:

```
oc rollout latest dc/<myapp>-kieserver
```

Replace **<myapp>** with the application name that was set when configuring the template.

3.4. MODIFYING THE TEMPLATE FOR THE SINGLE AUTHORING ENVIRONMENT

By default, the single authoring template uses the H2 database with permanent storage. If you prefer to create a MySQL or PostgreSQL pod or to use an external database server (outside the OpenShift project), you need to modify the template before deploying the environment.

An OpenShift template defines a set of objects that can be created by OpenShift. To change an environment configuration, you need to modify, add, or delete these objects. To simplify this task, comments are provided in the Red Hat Process Automation Manager templates.

Some comments mark blocks within the template, staring with **BEGIN** and ending with **END**. For example, the following block is named **Sample block**:

```
## Sample block BEGIN
sample line 1
sample line 2
sample line 3
## Sample block END
```

For some changes, you might need to replace a block in one template file with a block from another template file provided with Red Hat Process Automation Manager. In this case, delete the block, then paste the new block in its exact location.

Procedure

Edit the **rhpam75-authoring.yaml** template file to make any of the following changes as necessary.

- If you want to use MySQL instead of the H2 database, you need to replace several blocks of the file, marked with comments from **BEGIN** to **END**, with blocks from the **rhpam75-kieserver-mysql.yaml** file that are also marked with comments. You also need to remove several other blocks and to add blocks in designated locations:

  1. Replace the block named **H2 database parameters** with the block named **MySQL database parameters**. (Take this block and all subsequent replacement blocks from the **rhpam75-kieserver-mysql.yaml** file.)

  2. Replace the block named **H2 driver settings** with the block named **MySQL driver settings**.

  3. Replace the block named **H2 persistent volume claim** with the block named **MySQL persistent volume claim**.

  4. Remove the blocks named **H2 volume mount** and **H2 volume settings**.

  5. Under the comment **Place to add database service**, add the block named **MySQL service**.

  6. Under the comment **Place to add database deployment config**, add the block named **MySQL deployment config**.

- If you want to use PostgreSQL instead of the H2 database, you need to replace several blocks of the file, marked with comments from **BEGIN** to **END**, with blocks from the **rhpam75-kieserver-postgresql.yaml** file that are also marked with comments. You also need to remove several other blocks and to add blocks in designated locations:

  1. Replace the block named **H2 database parameters** with the block named **PostgreSQL database parameters**. (Take this block and all subsequent replacement blocks from the **rhpam75-kieserver-postgresql.yaml** file.)

  2. Replace the block named **H2 driver settings** with the block named **PostgreSQL driver settings**.

  3. Replace the block named **H2 persistent volume claim** with the block named **PostgreSQL persistent volume claim**.

  4. Remove the blocks named **H2 volume mount** and **H2 volume settings**.

  5. Under the comment **Place to add database service**, add the block named **PostgreSQL service**.

  6. Under the comment **Place to add database deployment config**, add the block named **PostgreSQL deployment config**.

- If you want to use an external database server, replace several blocks of the file, marked with comments from **BEGIN** to **END**, with blocks from the **rhpam75-kieserver-externaldb.yaml** file, and also remove some blocks:

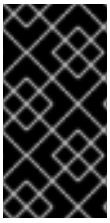
  1. Replace the block named **H2 database parameters** with the block named **External database parameters**. (Take this block and all subsequent replacement blocks from the **rhpam75-kieserver-externaldb.yaml** file.)

  2. Replace the block named **H2 driver settings** with the block named **External database driver settings**.

  3. Remove the following blocks of the file, marked with comments from **BEGIN** to **END**:

     - **H2 persistent volume claim**

- **H2 volume mount**
- **H2 volume settings**



IMPORTANT

The standard Process Server image includes drivers for MySQL and PostgreSQL external database servers. If you want to use another database server, you must build a custom Process Server image. For instructions, see Section 2.6, "Building a custom Process Server extension image for an external database".

3.5. MODIFYING THE TEMPLATE FOR THE HIGH AVAILABILITY AUTHORING ENVIRONMENT

By default, the high-availability authoring template creates a MySQL pod to provide the database server for the Process Server. If you prefer to use PostgreSQL or to use an external server (outside the OpenShift project), you need to modify the template before deploying the environment.

You can also modify the High Availability authoring template to change the number of replicas initially created for Business Central.

An OpenShift template defines a set of objects that can be created by OpenShift. To change an environment configuration, you need to modify, add, or delete these objects. To simplify this task, comments are provided in the Red Hat Process Automation Manager templates.

Some comments mark blocks within the template, staring with **BEGIN** and ending with **END**. For example, the following block is named **Sample block**:

```
## Sample block BEGIN
sample line 1
sample line 2
sample line 3
## Sample block END
```

For some changes, you might need to replace a block in one template file with a block from another template file provided with Red Hat Process Automation Manager. In this case, delete the block, then paste the new block in its exact location.

Procedure

Edit the **rhpam75-authoring-ha.yaml** template file to make any of the following changes as necessary.

- If you want to use PostgreSQL instead of MySQL, replace several blocks of the file, marked with comments from **BEGIN** to **END**, with blocks from the **rhpam75-kieserver-postgresql.yaml** file:
  1. Replace the block named **MySQL database parameters** with the block named **PosgreSQL database parameters**. (Take this block and all subsequent replacement blocks from the **rhpam75-kieserver-postgresql.yaml** file.)
  2. Replace the block named **MySQL service** with the block named **PosgrreSQL service**.
  3. Replace the block named **MySQL driver settings** with the block named **PosgreSQL driver settings**.
  4. Replace the block named **MySQL deployment config** with the block named **PosgreSQL deployment config**.

5. Replace the block named **MySQL persistent volume claim** with the block named **PosgreSQL persistent volume claim**.

- If you want to use an external database server, replace several blocks of the file, marked with comments from **BEGIN** to **END**, with blocks from the **rhpam75-kieserver-externaldb.yaml** file, and also remove some blocks:

  1. Replace the block named **MySQL database parameters** with the block named **External database parameters**. (Take this block and all subsequent replacement blocks from the **rhpam75-kieserver-externaldb.yaml** file.)

  2. Replace the block named **MySQL driver settings** with the block named **External database driver settings**.

  3. Remove the following blocks of the file, marked with comments from **BEGIN** to **END**:

     - **MySQL service**
     - **MySQL deployment config**
     - **MySQL persistent volume claim**

IMPORTANT

The standard Process Server image includes drivers for MySQL and PostgreSQL external database servers. If you want to use another database server, you must build a custom Process Server image. For instructions, see Section 2.6, “Building a custom Process Server extension image for an external database”.

- If you want to change the number of replicas initially created for Business Central, on the line below the comment **## Replicas for Business Central**, change the number of replicas to the desired value.

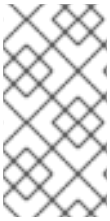
CHAPTER 4. RED HAT PROCESS AUTOMATION MANAGER ROLES AND USERS

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

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

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

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



NOTE

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

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

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

CHAPTER 5. OPENSHIFT TEMPLATE REFERENCE INFORMATION

Red Hat Process Automation Manager provides the following OpenShift templates. To access the templates, download and extract the **rhpam-7.5.1-openshift-templates.zip** product deliverable file from the Software Downloads page of the Red Hat customer portal.

- **rhpam75-authoring.yaml** provides a Business Central and a Process Server connected to the Business Central. The Process Server uses an H2 database with persistent storage. You can use this environment to author processes, services, and other business assets. For details about this template, see Section 5.1, "rhpam75-authoring.yaml template" .
- **rhpam75-authoring-ha.yaml** provides a high-availability Business Central, a Process Server connected to the Business Central, and a MySQL instance that the Process Server uses. You can use this environment to author processes, services, and other business assets. The high-availability functionality is in technology preview. For details about this template, see Section 5.2, "rhpam75-authoring-ha.yaml template".

5.1. RHPAM75-AUTHORING.YAML TEMPLATE

Application template for a non-HA persistent authoring environment, for Red Hat Process Automation Manager 7.5 - Deprecated

5.1.1. Parameters

Templates allow you to define parameters which take on a value. That value is then substituted wherever the parameter is referenced. References can be defined in any text field in the objects list field. Refer to the Openshift documentation for more information.

Variable name	Image Environment Variable	Description	Example value	Required
APPLICATION_NAME	–	The name for the application.	myapp	True
KIE_ADMIN_USER	**KIE_ADMIN_USER**	KIE administrator username.	adminUser	False
KIE_ADMIN_PWD	**KIE_ADMIN_PWD**	KIE administrator password.	–	False
KIE_SERVER_CONTROLLER_USER	**KIE_SERVER_CONTROLLER_USER**	KIE server controller username. (Sets the org.kie.server.controller.user system property)	controllerUser	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_CONTROLLER_PWD	**KIE_SERVER_CONTROLLER_PWD**	KIE server controller password. (Sets the org.kie.server.controller.pwd system property)	–	False
KIE_SERVER_CONTROLLER_TOKEN	**KIE_SERVER_CONTROLLER_TOKEN**	KIE server controller token for bearer authentication. (Sets the org.kie.server.controller.token system property)	–	False
KIE_SERVER_USER	**KIE_SERVER_USER**	KIE server username. (Sets the org.kie.server.user system property)	executionUser	False
KIE_SERVER_PWD	**KIE_SERVER_PWD**	KIE server password. (Sets the org.kie.server.pwd system property)	–	False
KIE_SERVER_BYPASS_AUTH_USER	**KIE_SERVER_BYPASS_AUTH_USER**	Allows the KIE server to bypass the authenticated user for task-related operations, for example, queries. (Sets the org.kie.server.bypass.auth.user system property)	false	False
KIE_SERVER_PERSISTENCE_DS	**RHPAM_JNDI**	KIE server persistence datasource. (Sets the org.kie.server.persistence.ds system property)	java:/jboss/datasources/rhpam	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_H2_USER	**RHPAM_USERNAME**	KIE server H2 database username.	sa	False
KIE_SERVER_H2_PWD	**RHPAM_PASSWORD**	KIE server H2 database password.	–	False
KIE_SERVER_MODE	**KIE_SERVER_MODE**	The KIE Server mode. Valid values are 'DEVELOPMENT' or 'PRODUCTION'. In production mode, you can not deploy SNAPSHOT versions of artifacts on the KIE server and can not change the version of an artifact in an existing container. (Sets the org.kie.server.mode system property)	**DEVELOPMENT**	False
KIE_MBEANS	**KIE_MBEANS**	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	enabled	False
DROOLS_SERVER_FILTER_CLASSES	**DROOLS_SERVER_FILTER_CLASSES**	KIE server class filtering. (Sets the org.drools.server.filter.classes system property)	true	False

Variable name	Image Environment Variable	Description	Example value	Required
PROMETHEUS_SERVER_EXT_DISABLED	**PROMETHEUS_SERVER_EXT_DISABLED**	If set to false, the prometheus server extension will be enabled. (Sets the org.kie.prometheus.server.ext.disabled system property)	false	False
BUSINESS_CENTRAL_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for the http service route for Business Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhpamcentr-<project>.<default-domain-suffix>	–	False
BUSINESS_CENTRAL_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for the https service route for Business Central. Leave blank for default hostname, e.g.: <application-name>-rhpamcentr-<project>.<default-domain-suffix>	–	False
KIE_SERVER_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for the http service route for KIE Server. Leave blank for default hostname, e.g.: insecure-<application-name>-kieserver-<project>.<default-domain-suffix>	–	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for the https service route for KIE Server. Leave blank for default hostname, e.g.: <application-name>-kieserver-<project>.<default-domain-suffix>	–	False
BUSINESS_CENTRAL_HTTPS_SECRET	–	The name of the secret containing the keystore file for Business Central.	businesscentral-app-secret	True
BUSINESS_CENTRAL_HTTPS_KEYSTORE	**HTTPS_KEYSTORE**	The name of the keystore file within the secret.	keystore.jks	False
BUSINESS_CENTRAL_HTTPS_NAME	**HTTPS_NAME**	The name associated with the server certificate.	jboss	False
BUSINESS_CENTRAL_HTTPS_PASSWORD	**HTTPS_PASSWORD**	The password for the keystore and certificate.	mykeystorepass	False
KIE_SERVER_HTTPS_SECRET	–	The name of the secret containing the keystore file for KIE server.	kieserver-app-secret	True
KIE_SERVER_HTTPS_KEYSTORE	**HTTPS_KEYSTORE**	The name of the keystore file within the secret.	keystore.jks	False
KIE_SERVER_HTTPS_NAME	**HTTPS_NAME**	The name associated with the server certificate.	jboss	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_HTTPS_PASSWORD	**HTTPS_PASSWORD**	The password for the keystore and certificate.	mykeystorepass	False
DB_VOLUME_CAPACITY	–	Size of persistent storage for the database volume.	1Gi	True
KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED	**KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED**	If set to true, turns on KIE server global discovery feature (Sets the org.kie.server.controller.openshift.global.discovery.enabled system property)	false	False
KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	**KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE**	If OpenShift integration of Business Central is turned on, setting this parameter to true enables connection to KIE Server via an OpenShift internal Service endpoint. (Sets the org.kie.server.controller.openshift.prefer.kieserver.service system property)	true	False
KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL	**KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL**	KIE ServerTemplate Cache TTL in milliseconds. (Sets the org.kie.server.controller.template.cache.ttl system property)	60000	False

Variable name	Image Environment Variable	Description	Example value	Required
IMAGE_STREAM_NAMESPACE	–	Namespace in which the ImageStreams for Red Hat Process Automation Manager images are installed. These ImageStreams are normally installed in the openshift namespace. You should only need to modify this if you installed the ImageStreams in a different namespace/project.	openshift	True
KIE_SERVER_IMAGE_STREAM_NAME	–	The name of the image stream to use for KIE server. Default is "rhpam-kieserver-rhel8".	rhpam-kieserver-rhel8	True
IMAGE_STREAM_TAG	–	A named pointer to an image in an image stream. Default is "7.5.0".	7.5.0	True
MAVEN_MIRROR_URL	**MAVEN_MIRROR_URL**	Maven mirror that Business Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	–	False
MAVEN_MIRROR_OF	**MAVEN_MIRROR_OF**	Maven mirror configuration for KIE server.	external:*,!repo-rhpamcentr	False

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_REPO_ID	**MAVEN_REPO_ID**	The id to use for the maven repository. If set, it can be excluded from the optionally configured mirror by adding it to MAVEN_MIRROR_OF. For example: external:*,!repo-rhpamcentr,!repo-custom. If MAVEN_MIRROR_URL is set but MAVEN_MIRROR_ID is not set, an id will be generated randomly, but won't be usable in MAVEN_MIRROR_OF.	repo-custom	False
MAVEN_REPO_URL	**MAVEN_REPO_URL**	Fully qualified URL to a Maven repository or service.	http://nexus.nexus-project.svc.cluster.local:8081/nexus/content/groups/public/	False
MAVEN_REPO_USERNAME	**MAVEN_REPO_USERNAME**	Username to access the Maven repository, if required.	–	False
MAVEN_REPO_PASSWORD	**MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	–	False
BUSINESS_CENTRAL_MAVEN_USERNAME	**KIE_MAVEN_USER**	Username to access the Maven service hosted by Business Central inside EAP.	mavenUser	True

Variable name	Image Environment Variable	Description	Example value	Required
BUSINESS_CENTRAL_MAVEN_PASSWORD	**KIE_MAVEN_PWD**	Password to access the Maven service hosted by Business Central inside EAP.	–	True
GIT_HOOKS_DIR	**GIT_HOOKS_DIR**	The directory to use for git hooks, if required.	**/opt/kie/data/git/hooks**	False
BUSINESS_CENTRAL_VOLUME_CAPACITY	–	Size of the persistent storage for Business Central runtime data.	1Gi	True
BUSINESS_CENTRAL_MEMORY_LIMIT	–	Business Central Container memory limit	2Gi	False
KIE_SERVER_MEMORY_LIMIT	–	KIE server Container memory limit	1Gi	False
SSO_URL	**SSO_URL**	RH-SSO URL.	https://rh-sso.example.com/auth	False
SSO_REALM	**SSO_REALM**	RH-SSO Realm name.	–	False
BUSINESS_CENTRAL_SSO_CLIENT	**SSO_CLIENT**	Business Central RH-SSO Client name.	–	False
BUSINESS_CENTRAL_SSO_SECRET	**SSO_SECRET**	Business Central RH-SSO Client Secret.	252793ed-7118-4ca8-8dab-5622fa97d892	False
KIE_SERVER_SSO_CLIENT	**SSO_CLIENT**	KIE Server RH-SSO Client name.	–	False
KIE_SERVER_SSO_SECRET	**SSO_SECRET**	KIE Server RH-SSO Client Secret.	252793ed-7118-4ca8-8dab-5622fa97d892	False

Variable name	Image Environment Variable	Description	Example value	Required
SSO_USERNAME	**SSO_USERNAME**	RH-SSO Realm Admin Username used to create the Client if it doesn't exist.	–	False
SSO_PASSWORD	**SSO_PASSWORD**	RH-SSO Realm Admin Password used to create the Client.	–	False
SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	**SSO_DISABLE_SSL_CERTIFICATE_VALIDATION**	RH-SSO Disable SSL Certificate Validation.	false	False
SSO_PRINCIPAL_ATTRIBUTE	**SSO_PRINCIPAL_ATTRIBUTE**	RH-SSO Principal Attribute to use as username.	preferred_username	False
AUTH_LDAP_URL	**AUTH_LDAP_URL**	LDAP Endpoint to connect for authentication	ldap://myldap.example.com	False
AUTH_LDAP_BIND_DN	**AUTH_LDAP_BIND_DN**	Bind DN used for authentication.	uid=admin,ou=users,ou=exmample,ou=com	False
AUTH_LDAP_BIND_CREDENTIAL	**AUTH_LDAP_BIND_CREDENTIAL**	LDAP Credentials used for authentication.	Password	False
AUTH_LDAP_JAAS_SECURITY_DOMAIN	**AUTH_LDAP_JAAS_SECURITY_DOMAIN**	The JMX ObjectName of the JaasSecurityDomain used to decrypt the password.	–	False
AUTH_LDAP_BASE_CTX_DN	**AUTH_LDAP_BASE_CTX_DN**	LDAP Base DN of the top-level context to begin the user search.	ou=users,ou=example,ou=com	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_BASE_FILTER	**AUTH_LDAP_BASE_FILTER**	LDAP search filter used to locate the context of the user to authenticate. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. A common example for the search filter is (uid={0}).	(uid={0})	False
AUTH_LDAP_SEARCH_SCOPE	**AUTH_LDAP_SEARCH_SCOPE**	The search scope to use.	**SUBTREE_SCOPE**	False
AUTH_LDAP_SEARCH_TIME_LIMIT	**AUTH_LDAP_SEARCH_TIME_LIMIT**	The timeout in milliseconds for user or role searches.	10000	False
AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	**AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE**	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	distinguishedName	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_PARSE_USERNAME	**AUTH_LDAP_PARSE_USERNAME**	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	true	False
AUTH_LDAP_USERNAME_BEGIN_STRING	**AUTH_LDAP_USERNAME_BEGIN_STRING**	Defines the String which is to be removed from the start of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	–	False
AUTH_LDAP_USERNAME_END_STRING	**AUTH_LDAP_USERNAME_END_STRING**	Defines the String which is to be removed from the end of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	–	False
AUTH_LDAP_ROLE_ATTRIBUTE_ID	**AUTH_LDAP_ROLE_ATTRIBUTE_ID**	Name of the attribute containing the user roles.	memberOf	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_ROLES_CTX_DN	**AUTH_LDAP_ROLES_CTX_DN**	The fixed DN of the context to search for user roles. This is not the DN where the actual roles are, but the DN where the objects containing the user roles are. For example, in a Microsoft Active Directory server, this is the DN where the user account is.	ou=groups,ou=example,ou=com	False
AUTH_LDAP_ROLE_FILTER	**AUTH_LDAP_ROLE_FILTER**	A search filter used to locate the roles associated with the authenticated user. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. The authenticated userDN is substituted into the filter anywhere a {1} is used. An example search filter that matches on the input username is (member={0}). An alternative that matches on the authenticated userDN is (member={1}).	(memberOf={1})	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_ROLE_RECURSION	**AUTH_LDAP_ROLE_RECURSION**	The number of levels of recursion the role search will go below a matching context. Disable recursion by setting this to 0.	1	False
AUTH_LDAP_DEFAULT_ROLE	**AUTH_LDAP_DEFAULT_ROLE**	A role included for all authenticated users	user	False
AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	**AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID**	Name of the attribute within the roleCtxDN context which contains the role name. If the roleAttributeIsDN property is set to true, this property is used to find the role object's name attribute.	name	False
AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	**AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN**	A flag indicating if the DN returned by a query contains the roleNameAttribute ID. If set to true, the DN is checked for the roleNameAttribute ID. If set to false, the DN is not checked for the roleNameAttribute ID. This flag can improve the performance of LDAP queries.	false	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	**AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN**	Whether or not the roleAttributeID contains the fully-qualified DN of a role object. If false, the role name is taken from the value of the roleNameAttributeId attribute of the context name. Certain directory schemas, such as Microsoft Active Directory, require this attribute to be set to true.	false	False
AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	**AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK**	If you are not using referrals, you can ignore this option. When using referrals, this option denotes the attribute name which contains users defined for a certain role, for example member, if the role object is inside the referral. Users are checked against the content of this attribute name. If this option is not set, the check will always fail, so role objects cannot be stored in a referral tree.	–	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_ROLE_MAPPER_ROLES_PROPERTIES	**AUTH_ROLE_MAPPER_ROLES_PROPERTIES**	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format is original_role=role1,role2,role3	–	False
AUTH_ROLE_MAPPER_REPLACE_ROLE	**AUTH_ROLE_MAPPER_REPLACE_ROLE**	Whether to add to the current roles, or replace the current roles with the mapped ones. Replaces if set to true.	–	False

5.1.2. Objects

The CLI supports various object types. A list of these object types as well as their abbreviations can be found in the Openshift documentation.

5.1.2.1. Services

A service is an abstraction which defines a logical set of pods and a policy by which to access them. Refer to the container-engine documentation for more information.

Service	Port	Name	Description
${APPLICATION_NAME}-rhpamcentr	8080	http	All the Business Central web server's ports.
	8443	https	
${APPLICATION_NAME}-kieserver	8080	http	All the KIE server web server's ports.
	8443	https	

5.1.2.2. Routes

A route is a way to expose a service by giving it an externally-reachable hostname such as **www.example.com**. A defined route and the endpoints identified by its service can be consumed by a router to provide named connectivity from external clients to your applications. Each route consists of a route name, service selector, and (optionally) security configuration. Refer to the Openshift documentation for more information.

Service	Security	Hostname
insecure-${APPLICATION_NAME}-rhpamcentr-http	none	**${BUSINESS_CENTRAL_HOSTNAME_HTTP}**
${APPLICATION_NAME}-rhpamcentr-https	TLS passthrough	**${BUSINESS_CENTRAL_HOSTNAME_HTTPS}**
insecure-${APPLICATION_NAME}-kieserver-http	none	**${KIE_SERVER_HOSTNAME_HTTP}**
${APPLICATION_NAME}-kieserver-https	TLS passthrough	**${KIE_SERVER_HOSTNAME_HTTPS}**

5.1.2.3. Deployment Configurations

A deployment in OpenShift is a replication controller based on a user defined template called a deployment configuration. Deployments are created manually or in response to triggered events. Refer to the Openshift documentation for more information.

5.1.2.3.1. Triggers

A trigger drives the creation of new deployments in response to events, both inside and outside OpenShift. Refer to the Openshift documentation for more information.

Deployment	Triggers
${APPLICATION_NAME}-rhpamcentr	ImageChange
${APPLICATION_NAME}-kieserver	ImageChange

5.1.2.3.2. Replicas

A replication controller ensures that a specified number of pod "replicas" are running at any one time. If there are too many, the replication controller kills some pods. If there are too few, it starts more. Refer to the container-engine documentation for more information.

Deployment	Replicas
${APPLICATION_NAME}-rhpamcentr	1

Deployment	Replicas
${APPLICATION_NAME}-kieserver	1

5.1.2.3.3. Pod Template

5.1.2.3.3.1. Service Accounts

Service accounts are API objects that exist within each project. They can be created or deleted like any other API object. Refer to the Openshift documentation for more information.

Deployment	Service Account
${APPLICATION_NAME}-rhpamcentr	**${APPLICATION_NAME}-rhpamsvc**
${APPLICATION_NAME}-kieserver	**${APPLICATION_NAME}-rhpamsvc**

5.1.2.3.3.2. Image

Deployment	Image
${APPLICATION_NAME}-rhpamcentr	rhpam-businesscentral-rhel8
${APPLICATION_NAME}-kieserver	**${KIE_SERVER_IMAGE_STREAM_NAME}**

5.1.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-rhpamcentr

> Http Get on http://localhost:8080/rest/ready

${APPLICATION_NAME}-kieserver

> Http Get on http://localhost:8080/services/rest/server/readycheck

5.1.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-rhpamcentr

> Http Get on http://localhost:8080/rest/healthy

${APPLICATION_NAME}-kieserver

> Http Get on http://localhost:8080/services/rest/server/healthcheck

5.1.2.3.3.5. Exposed Ports

Deployments	Name	Port	Protocol
${APPLICATION_NAME}-rhpamcentr	jolokia	8778	**TCP**
	http	8080	**TCP**
	https	8443	**TCP**
${APPLICATION_NAME}-kieserver	jolokia	8778	**TCP**
	http	8080	**TCP**
	https	8443	**TCP**

5.1.2.3.3.6. Image Environment Variables

Deployment	Variable name	Description	Example value
${APPLICATION_NAME}-rhpamcentr	**APPLICATION_USERS_PROPERTIES**	–	**/opt/kie/data/configuration/application-users.properties**
	APPLICATION_ROLES_PROPERTIES	–	**/opt/kie/data/configuration/application-roles.properties**
	KIE_ADMIN_USER	KIE administrator username.	**${KIE_ADMIN_USER}**
	KIE_ADMIN_PWD	KIE administrator password.	**${KIE_ADMIN_PWD}**
	KIE_MBEANS	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED	If set to true, turns on KIE server global discovery feature (Sets the org.kie.server.controller.openshift.global.discovery.enabled system property)	**${KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	If OpenShift integration of Business Central is turned on, setting this parameter to true enables connection to KIE Server via an OpenShift internal Service endpoint. (Sets the org.kie.server.controller.openshift.prefer.kieserver.service system property)	**${KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE}**
	KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL	KIE ServerTemplate Cache TTL in milliseconds. (Sets the org.kie.server.controller.template.cache.ttl system property)	**${KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL}**
	KIE_WORKBENCH_CONTROLLER_OPENSHIFT_ENABLED	–	true
	KIE_SERVER_CONTROLLER_USER	KIE server controller username. (Sets the org.kie.server.controller.user system property)	**${KIE_SERVER_CONTROLLER_USER}**
	KIE_SERVER_CONTROLLER_PWD	KIE server controller password. (Sets the org.kie.server.controller.pwd system property)	**${KIE_SERVER_CONTROLLER_PWD}**
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication. (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	KIE_SERVER_USER	KIE server username. (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_PWD	KIE server password. (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	MAVEN_MIRROR_URL	Maven mirror that Business Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	**${MAVEN_MIRROR_URL}**
	MAVEN_REPO_ID	The id to use for the maven repository. If set, it can be excluded from the optionally configured mirror by adding it to MAVEN_MIRROR_OF. For example: external:*,!repo-rhpamcentr,!repo-custom. If MAVEN_MIRROR_URL is set but MAVEN_MIRROR_ID is not set, an id will be generated randomly, but won't be usable in MAVEN_MIRROR_OF.	**${MAVEN_REPO_ID}**
	MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	KIE_MAVEN_USER	Username to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_USERNAME}**

Deployment	Variable name	Description	Example value
	KIE_MAVEN_PWD	Password to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_PASSWORD}**
	GIT_HOOKS_DIR	The directory to use for git hooks, if required.	**${GIT_HOOKS_DIR}**
	HTTPS_KEYSTORE_DIR	–	**/etc/businesscentral-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret.	**${BUSINESS_CENTRAL_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate.	**${BUSINESS_CENTRAL_HTTPS_NAME}**
	HTTPS_PASSWORD	The password for the keystore and certificate.	**${BUSINESS_CENTRAL_HTTPS_PASSWORD}**
	WORKBENCH_ROUTE_NAME	–	**${APPLICATION_NAME}-rhpamcentr**
	SSO_URL	RH-SSO URL.	**${SSO_URL}**
	SSO_OPENIDCONNECT_DEPLOYMENTS	–	ROOT.war
	SSO_REALM	RH-SSO Realm name.	**${SSO_REALM}**
	SSO_SECRET	Business Central RH-SSO Client Secret.	**${BUSINESS_CENTRAL_SSO_SECRET}**
	SSO_CLIENT	Business Central RH-SSO Client name.	**${BUSINESS_CENTRAL_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm Admin Username used to create the Client if it doesn't exist.	**${SSO_USERNAME}**
	SSO_PASSWORD	RH-SSO Realm Admin Password used to create the Client.	**${SSO_PASSWORD}**

Deployment	Variable name	Description	Example value
	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO Disable SSL Certificate Validation.	**${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}**
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as username.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for the http service route for Business Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhpamcentr-<project>.<default-domain-suffix>	**${BUSINESS_CENTRAL_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for the https service route for Business Central. Leave blank for default hostname, e.g.: <application-name>-rhpamcentr-<project>.<default-domain-suffix>	**${BUSINESS_CENTRAL_HOSTNAME_HTTPS}**
	AUTH_LDAP_URL	LDAP Endpoint to connect for authentication	**${AUTH_LDAP_URL}**
	AUTH_LDAP_BIND_DN	Bind DN used for authentication.	**${AUTH_LDAP_BIND_DN}**
	AUTH_LDAP_BIND_CREDENTIAL	LDAP Credentials used for authentication.	**${AUTH_LDAP_BIND_CREDENTIAL}**
	AUTH_LDAP_JAAS_SECURITY_DOMAIN	The JMX ObjectName of the JaasSecurityDomain used to decrypt the password.	**${AUTH_LDAP_JAAS_SECURITY_DOMAIN}**
	AUTH_LDAP_BASE_CTX_DN	LDAP Base DN of the top-level context to begin the user search.	**${AUTH_LDAP_BASE_CTX_DN}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_BASE_FILTER	LDAP search filter used to locate the context of the user to authenticate. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. A common example for the search filter is (uid={0}).	**${AUTH_LDAP_BASE_FILTER}**
	AUTH_LDAP_SEARCH_SCOPE	The search scope to use.	**${AUTH_LDAP_SEARCH_SCOPE}**
	AUTH_LDAP_SEARCH_TIME_LIMIT	The timeout in milliseconds for user or role searches.	**${AUTH_LDAP_SEARCH_TIME_LIMIT}**
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	**${AUTH_LDAP_PARSE_USERNAME}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_USERNAME_BEGIN_STRING	Defines the String which is to be removed from the start of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_END_STRING}**
	AUTH_LDAP_ROLE_ATTRIBUTE_ID	Name of the attribute containing the user roles.	**${AUTH_LDAP_ROLE_ATTRIBUTE_ID}**
	AUTH_LDAP_ROLES_CTX_DN	The fixed DN of the context to search for user roles. This is not the DN where the actual roles are, but the DN where the objects containing the user roles are. For example, in a Microsoft Active Directory server, this is the DN where the user account is.	**${AUTH_LDAP_ROLES_CTX_DN}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_ROLE_FILTER	A search filter used to locate the roles associated with the authenticated user. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. The authenticated userDN is substituted into the filter anywhere a {1} is used. An example search filter that matches on the input username is (member={0}). An alternative that matches on the authenticated userDN is (member={1}).	**${AUTH_LDAP_ROLE_FILTER}**
	AUTH_LDAP_ROLE_RECURSION	The number of levels of recursion the role search will go below a matching context. Disable recursion by setting this to 0.	**${AUTH_LDAP_ROLE_RECURSION}**
	AUTH_LDAP_DEFAULT_ROLE	A role included for all authenticated users	**${AUTH_LDAP_DEFAULT_ROLE}**
	AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	Name of the attribute within the roleCtxDN context which contains the role name. If the roleAttributeIsDN property is set to true, this property is used to find the role object's name attribute.	**${AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	A flag indicating if the DN returned by a query contains the roleNameAttributeID. If set to true, the DN is checked for the roleNameAttributeID. If set to false, the DN is not checked for the roleNameAttributeID. This flag can improve the performance of LDAP queries.	**${AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN}**
	AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	Whether or not the roleAttributeID contains the fully-qualified DN of a role object. If false, the role name is taken from the value of the roleNameAttributeId attribute of the context name. Certain directory schemas, such as Microsoft Active Directory, require this attribute to be set to true.	**${AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN}**
	AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	If you are not using referrals, you can ignore this option. When using referrals, this option denotes the attribute name which contains users defined for a certain role, for example member, if the role object is inside the referral. Users are checked against the content of this attribute name. If this option is not set, the check will always fail, so role objects cannot be stored in a referral tree.	**${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}**

Deployment	Variable name	Description	Example value
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format is original_role=role1,role2,role3	**${AUTH_ROLE_MAPPER_ROLES_PROPERTIES}**
	AUTH_ROLE_MAPPER_REPLACE_ROLE	Whether to add to the current roles, or replace the current roles with the mapped ones. Replaces if set to true.	**${AUTH_ROLE_MAPPER_REPLACE_ROLE}**
${APPLICATION_NAME}-kieserver	**WORKBENCH_SERVICE_NAME**	–	**${APPLICATION_NAME}-rhpamcentr**
	DATASOURCES	–	**RHPAM**
	RHPAM_DATABASE	–	rhpam7
	RHPAM_JNDI	KIE server persistence datasource. (Sets the org.kie.server.persistence.ds system property)	**${KIE_SERVER_PERSISTENCE_DS}**
	RHPAM_JTA	–	true
	RHPAM_DRIVER	–	h2
	RHPAM_USERNAME	KIE server H2 database username.	**${KIE_SERVER_H2_USER}**
	RHPAM_PASSWORD	KIE server H2 database password.	**${KIE_SERVER_H2_PWD}**
	RHPAM_NONXA	–	false
	RHPAM_XA_CONNECTION_PROPERTY_URL	–	jdbc:h2:/opt/kie/data/h2/rhpam;AUTO_SERVER=TRUE

Deployment	Variable name	Description	Example value
	KIE_SERVER_PERSISTENCE_DIALECT	–	org.hibernate.dialect.H2Dialect
	KIE_ADMIN_USER	KIE administrator username.	**${KIE_ADMIN_USER}**
	KIE_ADMIN_PWD	KIE administrator password.	**${KIE_ADMIN_PWD}**
	KIE_SERVER_MODE	The KIE Server mode. Valid values are 'DEVELOPMENT' or 'PRODUCTION'. In production mode, you can not deploy SNAPSHOT versions of artifacts on the KIE server and can not change the version of an artifact in an existing container. (Sets the org.kie.server.mode system property)	**${KIE_SERVER_MODE}**
	KIE_MBEANS	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	DROOLS_SERVER_FILTER_CLASSES	KIE server class filtering. (Sets the org.drools.server.filter.classes system property)	**${DROOLS_SERVER_FILTER_CLASSES}**
	PROMETHEUS_SERVER_EXT_DISABLED	If set to false, the prometheus server extension will be enabled. (Sets the org.kie.prometheus.server.ext.disabled system property)	**${PROMETHEUS_SERVER_EXT_DISABLED}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_BYPASS_AUTH_USER	Allows the KIE server to bypass the authenticated user for task-related operations, for example, queries. (Sets the org.kie.server.bypass.auth.user system property)	**${KIE_SERVER_BYPASS_AUTH_USER}**
	KIE_SERVER_ID	–	–
	KIE_SERVER_ROUTE_NAME	–	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_PERSISTENCE_DS	KIE server persistence datasource. (Sets the org.kie.server.persistence.ds system property)	**${KIE_SERVER_PERSISTENCE_DS}**
	KIE_SERVER_STARTUP_STRATEGY	–	OpenShiftStartupStrategy
	KIE_SERVER_USER	KIE server username. (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**
	KIE_SERVER_PWD	KIE server password. (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	MAVEN_MIRROR_URL	Maven mirror that Business Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	**${MAVEN_MIRROR_URL}**
	MAVEN_MIRROR_OF	Maven mirror configuration for KIE server.	**${MAVEN_MIRROR_OF}**
	MAVEN_REPOS	–	RHPAMCENTR,EXTERNAL

Deployment	Variable name	Description	Example value
	RHPAMCENTR_MAVEN_REPO_ID	–	repo-rhpamcentr
	RHPAMCENTR_MAVEN_REPO_SERVICE	–	**${APPLICATION_NAME}-rhpamcentr**
	RHPAMCENTR_MAVEN_REPO_PATH	–	**/maven2/**
	RHPAMCENTR_MAVEN_REPO_USERNAME	Username to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_USERNAME}**
	RHPAMCENTR_MAVEN_REPO_PASSWORD	Password to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_PASSWORD}**
	EXTERNAL_MAVEN_REPO_ID	The id to use for the maven repository. If set, it can be excluded from the optionally configured mirror by adding it to MAVEN_MIRROR_OF. For example: external:*,!repo-rhpamcentr,!repo-custom. If MAVEN_MIRROR_URL is set but MAVEN_MIRROR_ID is not set, an id will be generated randomly, but won't be usable in MAVEN_MIRROR_OF.	**${MAVEN_REPO_ID}**
	EXTERNAL_MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	EXTERNAL_MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	EXTERNAL_MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**

Deployment	Variable name	Description	Example value
	HTTPS_KEYSTORE_DIR	–	**/etc/kieserver-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret.	**${KIE_SERVER_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate.	**${KIE_SERVER_HTTPS_NAME}**
	HTTPS_PASSWORD	The password for the keystore and certificate.	**${KIE_SERVER_HTTPS_PASSWORD}**
	SSO_URL	RH-SSO URL.	**${SSO_URL}**
	SSO_OPENIDCONNECT_DEPLOYMENTS	–	ROOT.war
	SSO_REALM	RH-SSO Realm name.	**${SSO_REALM}**
	SSO_SECRET	KIE Server RH-SSO Client Secret.	**${KIE_SERVER_SSO_SECRET}**
	SSO_CLIENT	KIE Server RH-SSO Client name.	**${KIE_SERVER_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm Admin Username used to create the Client if it doesn't exist.	**${SSO_USERNAME}**
	SSO_PASSWORD	RH-SSO Realm Admin Password used to create the Client.	**${SSO_PASSWORD}**
	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO Disable SSL Certificate Validation.	**${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}**
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as username.	**${SSO_PRINCIPAL_ATTRIBUTE}**

Deployment	Variable name	Description	Example value
	HOSTNAME_HTTP	Custom hostname for the http service route for KIE Server. Leave blank for default hostname, e.g.: insecure-<application-name>-kieserver-<project>.<default-domain-suffix>	**${KIE_SERVER_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for the https service route for KIE Server. Leave blank for default hostname, e.g.: <application-name>-kieserver-<project>.<default-domain-suffix>	**${KIE_SERVER_HOSTNAME_HTTPS}**
	AUTH_LDAP_URL	LDAP Endpoint to connect for authentication	**${AUTH_LDAP_URL}**
	AUTH_LDAP_BIND_DN	Bind DN used for authentication.	**${AUTH_LDAP_BIND_DN}**
	AUTH_LDAP_BIND_CREDENTIAL	LDAP Credentials used for authentication.	**${AUTH_LDAP_BIND_CREDENTIAL}**
	AUTH_LDAP_JAAS_SECURITY_DOMAIN	The JMX ObjectName of the JaasSecurityDomain used to decrypt the password.	**${AUTH_LDAP_JAAS_SECURITY_DOMAIN}**
	AUTH_LDAP_BASE_CTX_DN	LDAP Base DN of the top-level context to begin the user search.	**${AUTH_LDAP_BASE_CTX_DN}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_BASE_FILTER	LDAP search filter used to locate the context of the user to authenticate. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. A common example for the search filter is (uid={0}).	**${AUTH_LDAP_BASE_FILTER}**
	AUTH_LDAP_SEARCH_SCOPE	The search scope to use.	**${AUTH_LDAP_SEARCH_SCOPE}**
	AUTH_LDAP_SEARCH_TIME_LIMIT	The timeout in milliseconds for user or role searches.	**${AUTH_LDAP_SEARCH_TIME_LIMIT}**
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	**${AUTH_LDAP_PARSE_USERNAME}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_USERNAME_BEGIN_STRING	Defines the String which is to be removed from the start of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_END_STRING}**
	AUTH_LDAP_ROLE_ATTRIBUTE_ID	Name of the attribute containing the user roles.	**${AUTH_LDAP_ROLE_ATTRIBUTE_ID}**
	AUTH_LDAP_ROLES_CTX_DN	The fixed DN of the context to search for user roles. This is not the DN where the actual roles are, but the DN where the objects containing the user roles are. For example, in a Microsoft Active Directory server, this is the DN where the user account is.	**${AUTH_LDAP_ROLES_CTX_DN}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_ROLE_FILTER	A search filter used to locate the roles associated with the authenticated user. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. The authenticated userDN is substituted into the filter anywhere a {1} is used. An example search filter that matches on the input username is (member={0}). An alternative that matches on the authenticated userDN is (member={1}).	**${AUTH_LDAP_ROLE_FILTER}**
	AUTH_LDAP_ROLE_RECURSION	The number of levels of recursion the role search will go below a matching context. Disable recursion by setting this to 0.	**${AUTH_LDAP_ROLE_RECURSION}**
	AUTH_LDAP_DEFAULT_ROLE	A role included for all authenticated users	**${AUTH_LDAP_DEFAULT_ROLE}**
	AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	Name of the attribute within the roleCtxDN context which contains the role name. If the roleAttributeIsDN property is set to true, this property is used to find the role object's name attribute.	**${AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	A flag indicating if the DN returned by a query contains the roleNameAttributeID. If set to true, the DN is checked for the roleNameAttributeID. If set to false, the DN is not checked for the roleNameAttributeID. This flag can improve the performance of LDAP queries.	**${AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN}**
	AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	Whether or not the roleAttributeID contains the fully-qualified DN of a role object. If false, the role name is taken from the value of the roleNameAttributeId attribute of the context name. Certain directory schemas, such as Microsoft Active Directory, require this attribute to be set to true.	**${AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN}**
	AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	If you are not using referrals, you can ignore this option. When using referrals, this option denotes the attribute name which contains users defined for a certain role, for example member, if the role object is inside the referral. Users are checked against the content of this attribute name. If this option is not set, the check will always fail, so role objects cannot be stored in a referral tree.	**${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}**

Deployment	Variable name	Description	Example value
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format is original_role=role1,role2,role3	**${AUTH_ROLE_MAPPER_ROLES_PROPERTIES}**
	AUTH_ROLE_MAPPER_REPLACE_ROLE	Whether to add to the current roles, or replace the current roles with the mapped ones. Replaces if set to true.	**${AUTH_ROLE_MAPPER_REPLACE_ROLE}**

5.1.2.3.3.7. Volumes

Deployment	Name	mountPath	Purpose	readOnly
${APPLICATION_NAME}-rhpamcentr	businesscentral-keystore-volume	**/etc/businesscentral-secret-volume**	ssl certs	True
${APPLICATION_NAME}-kieserver	kieserver-keystore-volume	**/etc/kieserver-secret-volume**	ssl certs	True

5.1.2.4. External Dependencies

5.1.2.4.1. Volume Claims

A **PersistentVolume** object is a storage resource in an OpenShift cluster. Storage is provisioned by an administrator by creating **PersistentVolume** objects from sources such as GCE Persistent Disks, AWS Elastic Block Stores (EBS), and NFS mounts. Refer to the Openshift documentation for more information.

Name	Access Mode
${APPLICATION_NAME}-rhpamcentr-claim	ReadWriteOnce
${APPLICATION_NAME}-kie-claim	ReadWriteMany

5.1.2.4.2. Secrets

This template requires the following secrets to be installed for the application to run.

businesscentral-app-secret kieserver-app-secret

5.2. RHPAM75-AUTHORING-HA.YAML TEMPLATE

Application template for a HA persistent authoring environment, for Red Hat Process Automation Manager 7.5 - Deprecated

5.2.1. Parameters

Templates allow you to define parameters which take on a value. That value is then substituted wherever the parameter is referenced. References can be defined in any text field in the objects list field. Refer to the Openshift documentation for more information.

Variable name	Image Environment Variable	Description	Example value	Required
APPLICATION_NAME	–	The name for the application.	myapp	True
KIE_ADMIN_USER	**KIE_ADMIN_USER**	KIE administrator username.	adminUser	False
KIE_ADMIN_PWD	**KIE_ADMIN_PWD**	KIE administrator password.	–	False
KIE_SERVER_CONTROLLER_USER	**KIE_SERVER_CONTROLLER_USER**	KIE server controller username. (Sets the org.kie.server.controller.user system property)	controllerUser	False
KIE_SERVER_CONTROLLER_PWD	**KIE_SERVER_CONTROLLER_PWD**	KIE server controller password. (Sets the org.kie.server.controller.pwd system property)	–	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_CONTROLLER_TOKEN	**KIE_SERVER_CONTROLLER_TOKEN**	KIE server controller token for bearer authentication. (Sets the org.kie.server.controller.token system property)	–	False
KIE_SERVER_USER	**KIE_SERVER_USER**	KIE server username. (Sets the org.kie.server.user system property)	executionUser	False
KIE_SERVER_PWD	**KIE_SERVER_PWD**	KIE server password. (Sets the org.kie.server.pwd system property)	–	False
KIE_SERVER_BYPASS_AUTH_USER	**KIE_SERVER_BYPASS_AUTH_USER**	Allows the KIE server to bypass the authenticated user for task-related operations, for example, queries. (Sets the org.kie.server.bypass.auth.user system property)	false	False
KIE_SERVER_PERSISTENCE_DS	**KIE_SERVER_PERSISTENCE_DS**	KIE server persistence datasource. (Sets the org.kie.server.persistence.ds system property)	java:/jboss/datasources/rhpam	False
MYSQL_USER	**RHPAM_USERNAME**	MySQL database username.	rhpam	False
MYSQL_PWD	**RHPAM_PASSWORD**	MySQL database password.	–	False
MYSQL_DB	**RHPAM_DATABASE**	MySQL database name.	rhpam7	False

Variable name	Image Environment Variable	Description	Example value	Required
MYSQL_DB_VOLUME_CAPACITY	–	Size of persistent storage for the KIE server database volume.	1Gi	True
MYSQL_IMAGE_STREAM_NAMESPACE	–	Namespace in which the ImageStream for the MySQL image is installed. The ImageStream is already installed in the openshift namespace. You should only need to modify this if you installed the ImageStream in a different namespace/project. Default is "openshift".	openshift	False
MYSQL_IMAGE_STREAM_TAG	–	The MySQL image version, which is intended to correspond to the MySQL version. Default is "5.7".	5.7	False
KIE_SERVER_MYSQL_DIALECT	**KIE_SERVER_PERSISTENCE_DIALECT**	KIE server MySQL Hibernate dialect.	org.hibernate.dialect.MySQL57Dialect	True

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_MODE	**KIE_SERVER_MODE**	The KIE Server mode. Valid values are 'DEVELOPMENT' or 'PRODUCTION'. In production mode, you can not deploy SNAPSHOT versions of artifacts on the KIE server and can not change the version of an artifact in an existing container. (Sets the org.kie.server.mode system property).	**DEVELOPMENT**	False
KIE_MBEANS	**KIE_MBEANS**	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	enabled	False
DROOLS_SERVER_FILTER_CLASSES	**DROOLS_SERVER_FILTER_CLASSES**	KIE server class filtering. (Sets the org.drools.server.filter.classes system property)	true	False
PROMETHEUS_SERVER_EXT_DISABLED	**PROMETHEUS_SERVER_EXT_DISABLED**	If set to false, the prometheus server extension will be enabled. (Sets the org.kie.prometheus.server.ext.disabled system property)	false	False

Variable name	Image Environment Variable	Description	Example value	Required
BUSINESS_CENTRAL_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route for Business Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhpamcentr-<project>.<default-domain-suffix>	–	False
BUSINESS_CENTRAL_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for https service route for Business Central. Leave blank for default hostname, e.g.: <application-name>-rhpamcentr-<project>.<default-domain-suffix>	–	False
KIE_SERVER_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route for KIE Server. Leave blank for default hostname, e.g.: insecure-<application-name>-kieserver-<project>.<default-domain-suffix>	–	False
KIE_SERVER_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for https service route for KIE Server. Leave blank for default hostname, e.g.: <application-name>-kieserver-<project>.<default-domain-suffix>	–	False

Variable name	Image Environment Variable	Description	Example value	Required
BUSINESS_CENTRAL_HTTPS_SECRET	–	The name of the secret containing the keystore file for Business Central.	businesscentral-app-secret	True
BUSINESS_CENTRAL_HTTPS_KEYSTORE	**HTTPS_KEYSTORE**	The name of the keystore file within the secret for Business Central.	keystore.jks	False
BUSINESS_CENTRAL_HTTPS_NAME	**HTTPS_NAME**	The name associated with the server certificate for Business Central.	jboss	False
BUSINESS_CENTRAL_HTTPS_PASSWORD	**HTTPS_PASSWORD**	The password for the keystore and certificate for Business Central.	mykeystorepass	False
KIE_SERVER_HTTPS_SECRET	–	The name of the secret containing the keystore file for KIE Server.	kieserver-app-secret	True
KIE_SERVER_HTTPS_KEYSTORE	**HTTPS_KEYSTORE**	The name of the keystore file within the secret for KIE Server.	keystore.jks	False
KIE_SERVER_HTTPS_NAME	**HTTPS_NAME**	The name associated with the server certificate for KIE Server.	jboss	False
KIE_SERVER_HTTPS_PASSWORD	**HTTPS_PASSWORD**	The password for the keystore and certificate for KIE Server.	mykeystorepass	False
APPFORMER_JMS_BROKER_USER	**APPFORMER_JMS_BROKER_USER**	The username to connect to the JMS broker.	jmsBrokerUser	True

Variable name	Image Environment Variable	Description	Example value	Required
APPFORMER_JMS_BROKER_PASSWORD	**APPFORMER_JMS_BROKER_PASSWORD**	The password to connect to the JMS broker.	–	True
DATAGRID_IMAGE	–	DataGrid image.	registry.redhat.io/jboss-datagrid-7/datagrid73-openshift:1.2	True
DATAGRID_CPU_LIMIT	–	DataGrid Container cpu limit.	1000m	True
DATAGRID_MEMORY_LIMIT	–	DataGrid Container memory limit	2Gi	True
DATAGRID_VOLUME_CAPACITY	–	Size of the persistent storage for DataGrid's runtime data.	1Gi	True
AMQ_BROKER_IMAGE	–	AMQ Broker Image	registry.redhat.io/amq7/amq-broker:7.4	True
AMQ_ROLE	–	User role for standard broker user.	admin	True
AMQ_NAME	–	The name of the broker	broker	True
AMQ_GLOBAL_MAX_SIZE	–	Specifies the maximum amount of memory that message data can consume. If no value is specified, half of the system's memory is allocated.	10 gb	False
AMQ_VOLUME_CAPACITY	–	Size of persistent storage for AMQ broker volume.	1Gi	True

Variable name	Image Environment Variable	Description	Example value	Required
AMQ_REPLICAS	–	Number of broker replicas for a cluster	2	True
KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED	**KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED**	If set to true, turns on KIE server global discovery feature (Sets the org.kie.server.controller.openshift.global.discovery.enabled system property)	false	False
KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	**KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE**	Enables connection to KIE Server via OpenShift internal Service endpoint (Sets the org.kie.server.controller.openshift.prefer.kieserver.service system property)	true	False
KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL	**KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL**	KIE ServerTemplate Cache TTL in milliseconds. (Sets the org.kie.server.controller.template.cache.ttl system property)	60000	False

Variable name	Image Environment Variable	Description	Example value	Required
IMAGE_STREAM_NAMESPACE	–	Namespace in which the ImageStreams for Red Hat Process Automation Manager images are installed. These ImageStreams are normally installed in the openshift namespace. You should only need to modify this if you installed the ImageStreams in a different namespace/project.	openshift	True
BUSINESS_CENTRAL_IMAGE_STREAM_NAME	–	The name of the image stream to use for Business Central. Default is "rhpam-businesscentral-rhel8".	rhpam-businesscentral-rhel8	True
KIE_SERVER_IMAGE_STREAM_NAME	–	The name of the image stream to use for KIE server. Default is "rhpam-kieserver-rhel8".	rhpam-kieserver-rhel8	True
IMAGE_STREAM_TAG	–	A named pointer to an image in an image stream. Default is "7.5.0".	7.5.0	True

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_MIRROR_URL	**MAVEN_MIRROR_URL**	Maven mirror that Business Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	–	False
MAVEN_MIRROR_OF	**MAVEN_MIRROR_OF**	Maven mirror configuration for KIE server.	external:*,!repo-rhpamcentr	False
MAVEN_REPO_ID	**MAVEN_REPO_ID**	The id to use for the maven repository. If set, it can be excluded from the optionally configured mirror by adding it to MAVEN_MIRROR_OF. For example: external:*,!repo-rhpamcentr,!repo-custom. If MAVEN_MIRROR_URL is set but MAVEN_MIRROR_ID is not set, an id will be generated randomly, but won't be usable in MAVEN_MIRROR_OF.	repo-custom	False
MAVEN_REPO_URL	**MAVEN_REPO_URL**	Fully qualified URL to a Maven repository or service.	http://nexus.nexus-project.svc.cluster.local:8081/nexus/content/groups/public/	False
MAVEN_REPO_USERNAME	**MAVEN_REPO_USERNAME**	Username to access the Maven repository, if required.	–	False

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_REPO_PASSWORD	**MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	–	False
BUSINESS_CENTRAL_MAVEN_USERNAME	**KIE_MAVEN_USER**	Username to access the Maven service hosted by Business Central inside EAP.	mavenUser	True
BUSINESS_CENTRAL_MAVEN_PASSWORD	**KIE_MAVEN_PWD**	Password to access the Maven service hosted by Business Central inside EAP.	–	True
GIT_HOOKS_DIR	**GIT_HOOKS_DIR**	The directory to use for git hooks, if required.	**/opt/kie/data/git/hooks**	False
TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL	**TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL**	Sets refresh-interval for the EJB timer database data-store service.	60000	True
BUSINESS_CENTRAL_VOLUME_CAPACITY	–	Size of the persistent storage for Business Central runtime data.	1Gi	True
BUSINESS_CENTRAL_MEMORY_LIMIT	–	Business Central Container memory limit.	2Gi	False
KIE_SERVER_MEMORY_LIMIT	–	KIE server Container memory limit.	1Gi	False
SSO_URL	**SSO_URL**	RH-SSO URL.	https://rh-sso.example.com/auth	False
SSO_REALM	**SSO_REALM**	RH-SSO Realm name.	–	False

Variable name	Image Environment Variable	Description	Example value	Required
BUSINESS_CENTRAL_SSO_CLIENT	**SSO_CLIENT**	Business Central RH-SSO Client name.	–	False
BUSINESS_CENTRAL_SSO_SECRET	**SSO_SECRET**	Business Central RH-SSO Client Secret.	252793ed-7118-4ca8-8dab-5622fa97d892	False
KIE_SERVER_SSO_CLIENT	**SSO_CLIENT**	KIE Server RH-SSO Client name.	–	False
KIE_SERVER_SSO_SECRET	**SSO_SECRET**	KIE Server RH-SSO Client Secret.	252793ed-7118-4ca8-8dab-5622fa97d892	False
SSO_USERNAME	**SSO_USERNAME**	RH-SSO Realm Admin Username used to create the Client if it doesn't exist.	–	False
SSO_PASSWORD	**SSO_PASSWORD**	RH-SSO Realm Admin Password used to create the Client.	–	False
SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	**SSO_DISABLE_SSL_CERTIFICATE_VALIDATION**	RH-SSO Disable SSL Certificate Validation.	false	False
SSO_PRINCIPAL_ATTRIBUTE	**SSO_PRINCIPAL_ATTRIBUTE**	RH-SSO Principal Attribute to use as username.	preferred_username	False
AUTH_LDAP_URL	**AUTH_LDAP_URL**	LDAP Endpoint to connect for authentication.	ldap://myldap.example.com	False
AUTH_LDAP_BIND_DN	**AUTH_LDAP_BIND_DN**	Bind DN used for authentication.	uid=admin,ou=users,ou=exmample,ou=com	False
AUTH_LDAP_BIND_CREDENTIAL	**AUTH_LDAP_BIND_CREDENTIAL**	LDAP Credentials used for authentication.	Password	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_JAAS_SECURITY_DOMAIN	**AUTH_LDAP_JAAS_SECURITY_DOMAIN**	The JMX ObjectName of the JaasSecurityDomain used to decrypt the password.	–	False
AUTH_LDAP_BASE_CTX_DN	**AUTH_LDAP_BASE_CTX_DN**	LDAP Base DN of the top-level context to begin the user search.	ou=users,ou=example,ou=com	False
AUTH_LDAP_BASE_FILTER	**AUTH_LDAP_BASE_FILTER**	LDAP search filter used to locate the context of the user to authenticate. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. A common example for the search filter is (uid={0}).	(uid={0})	False
AUTH_LDAP_SEARCH_SCOPE	**AUTH_LDAP_SEARCH_SCOPE**	The search scope to use.	**SUBTREE_SCOPE**	False
AUTH_LDAP_SEARCH_TIME_LIMIT	**AUTH_LDAP_SEARCH_TIME_LIMIT**	The timeout in milliseconds for user or role searches.	10000	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	**AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE**	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	distinguishedName	False
AUTH_LDAP_PARSE_USERNAME	**AUTH_LDAP_PARSE_USERNAME**	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	true	False
AUTH_LDAP_USERNAME_BEGIN_STRING	**AUTH_LDAP_USERNAME_BEGIN_STRING**	Defines the String which is to be removed from the start of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	–	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_USERNAME_END_STRING	**AUTH_LDAP_USERNAME_END_STRING**	Defines the String which is to be removed from the end of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	–	False
AUTH_LDAP_ROLE_ATTRIBUTE_ID	**AUTH_LDAP_ROLE_ATTRIBUTE_ID**	Name of the attribute containing the user roles.	memberOf	False
AUTH_LDAP_ROLES_CTX_DN	**AUTH_LDAP_ROLES_CTX_DN**	The fixed DN of the context to search for user roles. This is not the DN where the actual roles are, but the DN where the objects containing the user roles are. For example, in a Microsoft Active Directory server, this is the DN where the user account is.	ou=groups,ou=example,ou=com	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_ROLE_FILTER	**AUTH_LDAP_ROLE_FILTER**	A search filter used to locate the roles associated with the authenticated user. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. The authenticated userDN is substituted into the filter anywhere a {1} is used. An example search filter that matches on the input username is (member={0}). An alternative that matches on the authenticated userDN is (member={1}).	(memberOf={1})	False
AUTH_LDAP_ROLE_RECURSION	**AUTH_LDAP_ROLE_RECURSION**	The number of levels of recursion the role search will go below a matching context. Disable recursion by setting this to 0.	1	False
AUTH_LDAP_DEFAULT_ROLE	**AUTH_LDAP_DEFAULT_ROLE**	A role included for all authenticated users	user	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	**AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID**	Name of the attribute within the roleCtxDN context which contains the role name. If the roleAttributeIsDN property is set to true, this property is used to find the role object's name attribute.	name	False
AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	**AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN**	A flag indicating if the DN returned by a query contains the roleNameAttribute ID. If set to true, the DN is checked for the roleNameAttribute ID. If set to false, the DN is not checked for the roleNameAttribute ID. This flag can improve the performance of LDAP queries.	false	False
AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	**AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN**	Whether or not the roleAttributeID contains the fully-qualified DN of a role object. If false, the role name is taken from the value of the roleNameAttribute Id attribute of the context name. Certain directory schemas, such as Microsoft Active Directory, require this attribute to be set to true.	false	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	**AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK**	If you are not using referrals, you can ignore this option. When using referrals, this option denotes the attribute name which contains users defined for a certain role, for example member, if the role object is inside the referral. Users are checked against the content of this attribute name. If this option is not set, the check will always fail, so role objects cannot be stored in a referral tree.	–	False
AUTH_ROLE_MAPPER_ROLES_PROPERTIES	**AUTH_ROLE_MAPPER_ROLES_PROPERTIES**	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format of every entry in the file is original_role=role1,role2,role3	–	False

Variable name	Image Environment Variable	Description	Example value	Required
AUTH_ROLE_MAPPER_REPLACE_ROLE	**AUTH_ROLE_MAPPER_REPLACE_ROLE**	Whether to add to the current roles, or replace the current roles with the mapped ones. Replaces if set to true.	–	False

5.2.2. Objects

The CLI supports various object types. A list of these object types as well as their abbreviations can be found in the Openshift documentation.

5.2.2.1. Services

A service is an abstraction which defines a logical set of pods and a policy by which to access them. Refer to the container-engine documentation for more information.

Service	Port	Name	Description
${APPLICATION_NAME}-rhpamcentr	8080	http	All the Business Central web server's ports.
	8443	https	
${APPLICATION_NAME}-rhpamcentr-ping	8888	ping	The JGroups ping port for rhpamcentr clustering.
${APPLICATION_NAME}-datagrid-ping	8888	ping	The JGroups ping port for clustering.
${APPLICATION_NAME}-datagrid	11222	hotrod	Provides a service for accessing the application over Hot Rod protocol.
${APPLICATION_NAME}-kieserver	8080	http	All the KIE server web server's ports.
	8443	https	
${APPLICATION_NAME}-amq-tcp	61616	–	The broker's OpenWire port.
ping	8888	–	The JGroups ping port for amq clustering.

Service	Port	Name	Description
${APPLICATION_NAME}-mysql	3306	–	The MySQL server's port.

5.2.2.2. Routes

A route is a way to expose a service by giving it an externally-reachable hostname such as **www.example.com**. A defined route and the endpoints identified by its service can be consumed by a router to provide named connectivity from external clients to your applications. Each route consists of a route name, service selector, and (optionally) security configuration. Refer to the Openshift documentation for more information.

Service	Security	Hostname
insecure-${APPLICATION_NAME}-rhpamcentr-http	none	**${BUSINESS_CENTRAL_HOSTNAME_HTTP}**
${APPLICATION_NAME}-rhpamcentr-https	TLS passthrough	**${BUSINESS_CENTRAL_HOSTNAME_HTTPS}**
insecure-${APPLICATION_NAME}-kieserver-http	none	**${KIE_SERVER_HOSTNAME_HTTP}**
${APPLICATION_NAME}-kieserver-https	TLS passthrough	**${KIE_SERVER_HOSTNAME_HTTPS}**

5.2.2.3. Deployment Configurations

A deployment in OpenShift is a replication controller based on a user defined template called a deployment configuration. Deployments are created manually or in response to triggered events. Refer to the Openshift documentation for more information.

5.2.2.3.1. Triggers

A trigger drives the creation of new deployments in response to events, both inside and outside OpenShift. Refer to the Openshift documentation for more information.

Deployment	Triggers
${APPLICATION_NAME}-rhpamcentr	ImageChange
${APPLICATION_NAME}-kieserver	ImageChange
${APPLICATION_NAME}-mysql	ImageChange

5.2.2.3.2. Replicas

A replication controller ensures that a specified number of pod "replicas" are running at any one time. If there are too many, the replication controller kills some pods. If there are too few, it starts more. Refer to the container-engine documentation for more information.

Deployment	Replicas
${APPLICATION_NAME}-rhpamcentr	2
${APPLICATION_NAME}-kieserver	2
${APPLICATION_NAME}-mysql	1

5.2.2.3.3. Pod Template

5.2.2.3.3.1. Service Accounts

Service accounts are API objects that exist within each project. They can be created or deleted like any other API object. Refer to the Openshift documentation for more information.

Deployment	Service Account
${APPLICATION_NAME}-rhpamcentr	**${APPLICATION_NAME}-rhpamsvc**
${APPLICATION_NAME}-kieserver	**${APPLICATION_NAME}-rhpamsvc**

5.2.2.3.3.2. Image

Deployment	Image
${APPLICATION_NAME}-rhpamcentr	**${BUSINESS_CENTRAL_IMAGE_STREAM_NAME}**
${APPLICATION_NAME}-kieserver	**${KIE_SERVER_IMAGE_STREAM_NAME}**
${APPLICATION_NAME}-mysql	mysql

5.2.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-rhpamcentr

> Http Get on http://localhost:8080/rest/ready

${APPLICATION_NAME}-kieserver

> Http Get on http://localhost:8080/services/rest/server/readycheck

${APPLICATION_NAME}-mysql

```
/bin/sh -i -c MYSQL_PWD="$MYSQL_PASSWORD" mysql -h 127.0.0.1 -u $MYSQL_USER -D
$MYSQL_DATABASE -e 'SELECT 1'
```

5.2.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-rhpamcentr

```
Http Get on http://localhost:8080/rest/healthy
```

${APPLICATION_NAME}-kieserver

```
Http Get on http://localhost:8080/services/rest/server/healthcheck
```

${APPLICATION_NAME}-mysql

```
tcpSocket on port 3306
```

5.2.2.3.3.5. Exposed Ports

Deployments	Name	Port	Protocol
${APPLICATION_NAME}-rhpamcentr	jolokia	8778	**TCP**
	http	8080	**TCP**
	https	8443	**TCP**
	ping	8888	**TCP**
${APPLICATION_NAME}-kieserver	jolokia	8778	**TCP**
	http	8080	**TCP**
	https	8443	**TCP**
${APPLICATION_NAME}-mysql	–	3306	**TCP**

5.2.2.3.3.6. Image Environment Variables

Deployment	Variable name	Description	Example value
${APPLICATION_NAME}-rhpamcentr	**APPLICATION_USERS_PROPERTIES**	–	**/opt/kie/data/configuration/application-users.properties**

Deployment	Variable name	Description	Example value
	APPLICATION_ROLES_PROPERTIES	–	**/opt/kie/data/configuration/application-roles.properties**
	KIE_ADMIN_USER	KIE administrator username.	**${KIE_ADMIN_USER}**
	KIE_ADMIN_PWD	KIE administrator password.	**${KIE_ADMIN_PWD}**
	KIE_MBEANS	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED	If set to true, turns on KIE server global discovery feature (Sets the org.kie.server.controller.openshift.global.discovery.enabled system property)	**${KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED}**
	KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	Enables connection to KIE Server via OpenShift internal Service endpoint (Sets the org.kie.server.controller.openshift.prefer.kieserver.service system property)	**${KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE}**
	KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL	KIE ServerTemplate Cache TTL in milliseconds. (Sets the org.kie.server.controller.template.cache.ttl system property)	**${KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL}**
	KIE_WORKBENCH_CONTROLLER_OPENSHIFT_ENABLED	–	true
	KIE_SERVER_CONTROLLER_USER	KIE server controller username. (Sets the org.kie.server.controller.user system property)	**${KIE_SERVER_CONTROLLER_USER}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_CONTROLLER_PWD	KIE server controller password. (Sets the org.kie.server.controller.pwd system property)	**${KIE_SERVER_CONTROLLER_PWD}**
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication. (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	KIE_SERVER_USER	KIE server username. (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**
	KIE_SERVER_PWD	KIE server password. (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	WORKBENCH_ROUTE_NAME	–	**${APPLICATION_NAME}-rhpamcentr**
	MAVEN_MIRROR_URL	Maven mirror that Business Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	**${MAVEN_MIRROR_URL}**

Deployment	Variable name	Description	Example value
	MAVEN_REPO_ID	The id to use for the maven repository. If set, it can be excluded from the optionally configured mirror by adding it to MAVEN_MIRROR_OF. For example: external:*,!repo-rhpamcentr,!repo-custom. If MAVEN_MIRROR_URL is set but MAVEN_MIRROR_ID is not set, an id will be generated randomly, but won't be usable in MAVEN_MIRROR_OF.	**${MAVEN_REPO_ID}**
	MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	KIE_MAVEN_USER	Username to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_USERNAME}**
	KIE_MAVEN_PWD	Password to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_PASSWORD}**
	GIT_HOOKS_DIR	The directory to use for git hooks, if required.	**${GIT_HOOKS_DIR}**
	HTTPS_KEYSTORE_DIR	–	**/etc/businesscentral-secret-volume**

Deployment	Variable name	Description	Example value
	HTTPS_KEYSTORE	The name of the keystore file within the secret for Business Central.	**${BUSINESS_CENTRAL_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate for Business Central.	**${BUSINESS_CENTRAL_HTTPS_NAME}**
	HTTPS_PASSWORD	The password for the keystore and certificate for Business Central.	**${BUSINESS_CENTRAL_HTTPS_PASSWORD}**
	JGROUPS_PING_PROTOCOL	–	openshift.DNS_PING
	OPENSHIFT_DNS_PING_SERVICE_NAME	–	**${APPLICATION_NAME}-rhpamcentr-ping**
	OPENSHIFT_DNS_PING_SERVICE_PORT	–	8888
	APPFORMER_INFINISPAN_SERVICE_NAME	–	**${APPLICATION_NAME}-datagrid**
	APPFORMER_INFINISPAN_PORT	–	11222
	APPFORMER_JMS_BROKER_ADDRESS	–	**${APPLICATION_NAME}-amq-tcp**
	APPFORMER_JMS_BROKER_PORT	–	61616
	APPFORMER_JMS_BROKER_USER	The username to connect to the JMS broker.	**${APPFORMER_JMS_BROKER_USER}**
	APPFORMER_JMS_BROKER_PASSWORD	The password to connect to the JMS broker.	**${APPFORMER_JMS_BROKER_PASSWORD}**
	SSO_URL	RH-SSO URL.	**${SSO_URL}**

Deployment	Variable name	Description	Example value
	SSO_OPENIDCONNECT_DEPLOYMENTS	–	ROOT.war
	SSO_REALM	RH-SSO Realm name.	**${SSO_REALM}**
	SSO_SECRET	Business Central RH-SSO Client Secret.	**${BUSINESS_CENTRAL_SSO_SECRET}**
	SSO_CLIENT	Business Central RH-SSO Client name.	**${BUSINESS_CENTRAL_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm Admin Username used to create the Client if it doesn't exist.	**${SSO_USERNAME}**
	SSO_PASSWORD	RH-SSO Realm Admin Password used to create the Client.	**${SSO_PASSWORD}**
	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO Disable SSL Certificate Validation.	**${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}**
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as username.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route for Business Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhpamcentr-<project>.<default-domain-suffix>	**${BUSINESS_CENTRAL_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for https service route for Business Central. Leave blank for default hostname, e.g.: <application-name>-rhpamcentr-<project>.<default-domain-suffix>	**${BUSINESS_CENTRAL_HOSTNAME_HTTPS}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_URL	LDAP Endpoint to connect for authentication.	**${AUTH_LDAP_URL}**
	AUTH_LDAP_BIND_DN	Bind DN used for authentication.	**${AUTH_LDAP_BIND_DN}**
	AUTH_LDAP_BIND_CREDENTIAL	LDAP Credentials used for authentication.	**${AUTH_LDAP_BIND_CREDENTIAL}**
	AUTH_LDAP_JAAS_SECURITY_DOMAIN	The JMX ObjectName of the JaasSecurityDomain used to decrypt the password.	**${AUTH_LDAP_JAAS_SECURITY_DOMAIN}**
	AUTH_LDAP_BASE_CTX_DN	LDAP Base DN of the top-level context to begin the user search.	**${AUTH_LDAP_BASE_CTX_DN}**
	AUTH_LDAP_BASE_FILTER	LDAP search filter used to locate the context of the user to authenticate. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. A common example for the search filter is (uid={0}).	**${AUTH_LDAP_BASE_FILTER}**
	AUTH_LDAP_SEARCH_SCOPE	The search scope to use.	**${AUTH_LDAP_SEARCH_SCOPE}**
	AUTH_LDAP_SEARCH_TIME_LIMIT	The timeout in milliseconds for user or role searches.	**${AUTH_LDAP_SEARCH_TIME_LIMIT}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	**${AUTH_LDAP_PARSE_USERNAME}**
	AUTH_LDAP_USERNAME_BEGIN_STRING	Defines the String which is to be removed from the start of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_END_STRING}**
	AUTH_LDAP_ROLE_ATTRIBUTE_ID	Name of the attribute containing the user roles.	**${AUTH_LDAP_ROLE_ATTRIBUTE_ID}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_ROLES_CTX_DN	The fixed DN of the context to search for user roles. This is not the DN where the actual roles are, but the DN where the objects containing the user roles are. For example, in a Microsoft Active Directory server, this is the DN where the user account is.	**${AUTH_LDAP_ROLES_CTX_DN}**
	AUTH_LDAP_ROLE_FILTER	A search filter used to locate the roles associated with the authenticated user. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. The authenticated userDN is substituted into the filter anywhere a {1} is used. An example search filter that matches on the input username is (member={0}). An alternative that matches on the authenticated userDN is (member={1}).	**${AUTH_LDAP_ROLE_FILTER}**
	AUTH_LDAP_ROLE_RECURSION	The number of levels of recursion the role search will go below a matching context. Disable recursion by setting this to 0.	**${AUTH_LDAP_ROLE_RECURSION}**
	AUTH_LDAP_DEFAULT_ROLE	A role included for all authenticated users	**${AUTH_LDAP_DEFAULT_ROLE}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	Name of the attribute within the roleCtxDN context which contains the role name. If the roleAttributeIsDN property is set to true, this property is used to find the role object's name attribute.	**${AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID}**
	AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	A flag indicating if the DN returned by a query contains the roleNameAttributeID. If set to true, the DN is checked for the roleNameAttributeID. If set to false, the DN is not checked for the roleNameAttributeID. This flag can improve the performance of LDAP queries.	**${AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN}**
	AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	Whether or not the roleAttributeID contains the fully-qualified DN of a role object. If false, the role name is taken from the value of the roleNameAttributeId attribute of the context name. Certain directory schemas, such as Microsoft Active Directory, require this attribute to be set to true.	**${AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	If you are not using referrals, you can ignore this option. When using referrals, this option denotes the attribute name which contains users defined for a certain role, for example member, if the role object is inside the referral. Users are checked against the content of this attribute name. If this option is not set, the check will always fail, so role objects cannot be stored in a referral tree.	**${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}**
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format of every entry in the file is original_role=role1,role2,role3	**${AUTH_ROLE_MAPPER_ROLES_PROPERTIES}**
	AUTH_ROLE_MAPPER_REPLACE_ROLE	Whether to add to the current roles, or replace the current roles with the mapped ones. Replaces if set to true.	**${AUTH_ROLE_MAPPER_REPLACE_ROLE}**
${APPLICATION_NAME}-kieserver	**WORKBENCH_SERVICE_NAME**	–	**${APPLICATION_NAME}-rhpamcentr**
	TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL	Sets refresh-interval for the EJB timer database data-store service.	**${TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL}**
	DATASOURCES	–	**RHPAM**

Deployment	Variable name	Description	Example value
	RHPAM_DATABASE	MySQL database name.	**${MYSQL_DB}**
	RHPAM_DRIVER	–	mariadb
	RHPAM_USERNAME	MySQL database username.	**${MYSQL_USER}**
	RHPAM_PASSWORD	MySQL database password.	**${MYSQL_PWD}**
	RHPAM_SERVICE_HOST	–	**${APPLICATION_NAME}-mysql**
	RHPAM_SERVICE_PORT	–	3306
	KIE_SERVER_PERSISTENCE_DIALECT	KIE server MySQL Hibernate dialect.	**${KIE_SERVER_MYSQL_DIALECT}**
	KIE_SERVER_PERSISTENCE_DS	KIE server persistence datasource. (Sets the org.kie.server.persistence.ds system property)	**${KIE_SERVER_PERSISTENCE_DS}**
	RHPAM_JNDI	KIE server persistence datasource. (Sets the org.kie.server.persistence.ds system property)	**${KIE_SERVER_PERSISTENCE_DS}**
	RHPAM_JTA	–	true
	KIE_ADMIN_USER	KIE administrator username.	**${KIE_ADMIN_USER}**
	KIE_ADMIN_PWD	KIE administrator password.	**${KIE_ADMIN_PWD}**
	KIE_MBEANS	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_MODE	The KIE Server mode. Valid values are 'DEVELOPMENT' or 'PRODUCTION'. In production mode, you can not deploy SNAPSHOT versions of artifacts on the KIE server and can not change the version of an artifact in an existing container. (Sets the org.kie.server.mode system property).	**${KIE_SERVER_MODE}**
	DROOLS_SERVER_FILTER_CLASSES	KIE server class filtering. (Sets the org.drools.server.filter.classes system property)	**${DROOLS_SERVER_FILTER_CLASSES}**
	PROMETHEUS_SERVER_EXT_DISABLED	If set to false, the prometheus server extension will be enabled. (Sets the org.kie.prometheus.server.ext.disabled system property)	**${PROMETHEUS_SERVER_EXT_DISABLED}**
	KIE_SERVER_BYPASS_AUTH_USER	Allows the KIE server to bypass the authenticated user for task-related operations, for example, queries. (Sets the org.kie.server.bypass.auth.user system property)	**${KIE_SERVER_BYPASS_AUTH_USER}**
	KIE_SERVER_ID	–	–
	KIE_SERVER_ROUTE_NAME	–	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_STARTUP_STRATEGY	–	OpenShiftStartupStrategy
	KIE_SERVER_PWD	KIE server password. (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_USER	KIE server username. (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**
	MAVEN_MIRROR_URL	Maven mirror that Business Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	**${MAVEN_MIRROR_URL}**
	MAVEN_MIRROR_OF	Maven mirror configuration for KIE server.	**${MAVEN_MIRROR_OF}**
	MAVEN_REPOS	–	RHPAMCENTR,EXTERNAL
	RHPAMCENTR_MAVEN_REPO_ID	–	repo-rhpamcentr
	RHPAMCENTR_MAVEN_REPO_SERVICE	–	**${APPLICATION_NAME}-rhpamcentr**
	RHPAMCENTR_MAVEN_REPO_PATH	–	**/maven2/**
	RHPAMCENTR_MAVEN_REPO_USERNAME	Username to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_USERNAME}**
	RHPAMCENTR_MAVEN_REPO_PASSWORD	Password to access the Maven service hosted by Business Central inside EAP.	**${BUSINESS_CENTRAL_MAVEN_PASSWORD}**

Deployment	Variable name	Description	Example value
	EXTERNAL_MAVEN_REPO_ID	The id to use for the maven repository. If set, it can be excluded from the optionally configured mirror by adding it to MAVEN_MIRROR_OF. For example: external:*,!repo-rhpamcentr,!repo-custom. If MAVEN_MIRROR_URL is set but MAVEN_MIRROR_ID is not set, an id will be generated randomly, but won't be usable in MAVEN_MIRROR_OF.	**${MAVEN_REPO_ID}**
	EXTERNAL_MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	EXTERNAL_MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	EXTERNAL_MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	HTTPS_KEYSTORE_DIR	–	**/etc/kieserver-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret for KIE Server.	**${KIE_SERVER_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate for KIE Server.	**${KIE_SERVER_HTTPS_NAME}**
	HTTPS_PASSWORD	The password for the keystore and certificate for KIE Server.	**${KIE_SERVER_HTTPS_PASSWORD}**
	SSO_URL	RH-SSO URL.	**${SSO_URL}**

Deployment	Variable name	Description	Example value
	SSO_OPENIDCONNECT_DEPLOYMENTS	–	ROOT.war
	SSO_REALM	RH-SSO Realm name.	**${SSO_REALM}**
	SSO_SECRET	KIE Server RH-SSO Client Secret.	**${KIE_SERVER_SSO_SECRET}**
	SSO_CLIENT	KIE Server RH-SSO Client name.	**${KIE_SERVER_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm Admin Username used to create the Client if it doesn't exist.	**${SSO_USERNAME}**
	SSO_PASSWORD	RH-SSO Realm Admin Password used to create the Client.	**${SSO_PASSWORD}**
	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO Disable SSL Certificate Validation.	**${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}**
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as username.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route for KIE Server. Leave blank for default hostname, e.g.: insecure-<application-name>-kieserver-<project>.<default-domain-suffix>	**${KIE_SERVER_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for https service route for KIE Server. Leave blank for default hostname, e.g.: <application-name>-kieserver-<project>.<default-domain-suffix>	**${KIE_SERVER_HOSTNAME_HTTPS}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_URL	LDAP Endpoint to connect for authentication.	**${AUTH_LDAP_URL}**
	AUTH_LDAP_BIND_DN	Bind DN used for authentication.	**${AUTH_LDAP_BIND_DN}**
	AUTH_LDAP_BIND_CREDENTIAL	LDAP Credentials used for authentication.	**${AUTH_LDAP_BIND_CREDENTIAL}**
	AUTH_LDAP_JAAS_SECURITY_DOMAIN	The JMX ObjectName of the JaasSecurityDomain used to decrypt the password.	**${AUTH_LDAP_JAAS_SECURITY_DOMAIN}**
	AUTH_LDAP_BASE_CTX_DN	LDAP Base DN of the top-level context to begin the user search.	**${AUTH_LDAP_BASE_CTX_DN}**
	AUTH_LDAP_BASE_FILTER	LDAP search filter used to locate the context of the user to authenticate. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. A common example for the search filter is (uid={0}).	**${AUTH_LDAP_BASE_FILTER}**
	AUTH_LDAP_SEARCH_SCOPE	The search scope to use.	**${AUTH_LDAP_SEARCH_SCOPE}**
	AUTH_LDAP_SEARCH_TIME_LIMIT	The timeout in milliseconds for user or role searches.	**${AUTH_LDAP_SEARCH_TIME_LIMIT}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	**${AUTH_LDAP_PARSE_USERNAME}**
	AUTH_LDAP_USERNAME_BEGIN_STRING	Defines the String which is to be removed from the start of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_END_STRING}**
	AUTH_LDAP_ROLE_ATTRIBUTE_ID	Name of the attribute containing the user roles.	**${AUTH_LDAP_ROLE_ATTRIBUTE_ID}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_ROLES_CTX_DN	The fixed DN of the context to search for user roles. This is not the DN where the actual roles are, but the DN where the objects containing the user roles are. For example, in a Microsoft Active Directory server, this is the DN where the user account is.	**${AUTH_LDAP_ROLES_CTX_DN}**
	AUTH_LDAP_ROLE_FILTER	A search filter used to locate the roles associated with the authenticated user. The input username or userDN obtained from the login module callback is substituted into the filter anywhere a {0} expression is used. The authenticated userDN is substituted into the filter anywhere a {1} is used. An example search filter that matches on the input username is (member={0}). An alternative that matches on the authenticated userDN is (member={1}).	**${AUTH_LDAP_ROLE_FILTER}**
	AUTH_LDAP_ROLE_RECURSION	The number of levels of recursion the role search will go below a matching context. Disable recursion by setting this to 0.	**${AUTH_LDAP_ROLE_RECURSION}**
	AUTH_LDAP_DEFAULT_ROLE	A role included for all authenticated users	**${AUTH_LDAP_DEFAULT_ROLE}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	Name of the attribute within the roleCtxDN context which contains the role name. If the roleAttributeIsDN property is set to true, this property is used to find the role object's name attribute.	**${AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID}**
	AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	A flag indicating if the DN returned by a query contains the roleNameAttributeID. If set to true, the DN is checked for the roleNameAttributeID. If set to false, the DN is not checked for the roleNameAttributeID. This flag can improve the performance of LDAP queries.	**${AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN}**
	AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	Whether or not the roleAttributeID contains the fully-qualified DN of a role object. If false, the role name is taken from the value of the roleNameAttributeId attribute of the context name. Certain directory schemas, such as Microsoft Active Directory, require this attribute to be set to true.	**${AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	If you are not using referrals, you can ignore this option. When using referrals, this option denotes the attribute name which contains users defined for a certain role, for example member, if the role object is inside the referral. Users are checked against the content of this attribute name. If this option is not set, the check will always fail, so role objects cannot be stored in a referral tree.	**${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}**
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format of every entry in the file is original_role=role1,role2,role3	**${AUTH_ROLE_MAPPER_ROLES_PROPERTIES}**
	AUTH_ROLE_MAPPER_REPLACE_ROLE	Whether to add to the current roles, or replace the current roles with the mapped ones. Replaces if set to true.	**${AUTH_ROLE_MAPPER_REPLACE_ROLE}**
${APPLICATION_NAME}-mysql	**MYSQL_USER**	MySQL database username.	**${MYSQL_USER}**
	MYSQL_PASSWORD	MySQL database password.	**${MYSQL_PWD}**
	MYSQL_DATABASE	MySQL database name.	**${MYSQL_DB}**

5.2.2.3.3.7. Volumes

Deployment	Name	mountPath	Purpose	readOnly
${APPLICATION_NAME}-rhpamcentr	businesscentral-keystore-volume	**/etc/businesscentral-secret-volume**	ssl certs	True
${APPLICATION_NAME}-kieserver	kieserver-keystore-volume	**/etc/kieserver-secret-volume**	ssl certs	True
${APPLICATION_NAME}-mysql	**${APPLICATION_NAME}-mysql-pvol**	**/var/lib/mysql/data**	mysql	false

5.2.2.4. External Dependencies

5.2.2.4.1. Volume Claims

A **PersistentVolume** object is a storage resource in an OpenShift cluster. Storage is provisioned by an administrator by creating **PersistentVolume** objects from sources such as GCE Persistent Disks, AWS Elastic Block Stores (EBS), and NFS mounts. Refer to the Openshift documentation for more information.

Name	Access Mode
${APPLICATION_NAME}-rhpamcentr-claim	ReadWriteMany
${APPLICATION_NAME}-mysql-claim	ReadWriteOnce

5.2.2.4.2. Secrets

This template requires the following secrets to be installed for the application to run.

businesscentral-app-secret kieserver-app-secret

5.2.2.4.3. Clustering

Clustering in OpenShift EAP is achieved through one of two discovery mechanisms: Kubernetes or DNS. This is done by configuring the JGroups protocol stack in standalone-openshift.xml with either the **<openshift.KUBE_PING/>** or **<openshift.DNS_PING/>** elements. The templates are configured to use **DNS_PING**, however `KUBE_PING`is the default used by the image.

The discovery mechanism used is specified by the **JGROUPS_PING_PROTOCOL** environment variable which can be set to either **openshift.DNS_PING** or **openshift.KUBE_PING**. **openshift.KUBE_PING** is the default used by the image if no value is specified for **JGROUPS_PING_PROTOCOL**.

For DNS_PING to work, the following steps must be taken:

1. The **OPENSHIFT_DNS_PING_SERVICE_NAME** environment variable must be set to the name of the ping service for the cluster (see table above). If not set, the server will act as if it is a single-node cluster (a "cluster of one").

2. The **OPENSHIFT_DNS_PING_SERVICE_PORT** environment variables should be set to the port number on which the ping service is exposed (see table above). The **DNS_PING** protocol will attempt to discern the port from the SRV records, if it can, otherwise it will default to 8888.

3. A ping service which exposes the ping port must be defined. This service should be "headless" (ClusterIP=None) and must have the following:

    a. The port must be named for port discovery to work.

    b. It must be annotated with **service.alpha.kubernetes.io/tolerate-unready-endpoints** set to **"true"**. Omitting this annotation will result in each node forming their own "cluster of one" during startup, then merging their cluster into the other nodes' clusters after startup (as the other nodes are not detected until after they have started).

Example ping service for use with DNS_PING

```
kind: Service
apiVersion: v1
spec:
   clusterIP: None
   ports:
   - name: ping
     port: 8888
   selector:
      deploymentConfig: eap-app
metadata:
   name: eap-app-ping
   annotations:
      service.alpha.kubernetes.io/tolerate-unready-endpoints: "true"
      description: "The JGroups ping port for clustering."
```

For **KUBE_PING** to work, the following steps must be taken:

1. The **OPENSHIFT_KUBE_PING_NAMESPACE** environment variable must be set (see table above). If not set, the server will act as if it is a single-node cluster (a "cluster of one").

2. The **OPENSHIFT_KUBE_PING_LABELS** environment variables should be set (see table above). If not set, pods outside of your application (albeit in your namespace) will try to join.

3. Authorization must be granted to the service account the pod is running under to be allowed to access Kubernetes' REST api. This is done on the command line.

Example 5.1. Policy commands

Using the default service account in the myproject namespace:

```
oc policy add-role-to-user view system:serviceaccount:myproject:default -n myproject
```

Using the eap-service-account in the myproject namespace:

```
oc policy add-role-to-user view system:serviceaccount:myproject:eap-service-account -n
myproject
```

5.3. OPENSHIFT USAGE QUICK REFERENCE

To deploy, monitor, manage, and undeploy Red Hat Process Automation Manager templates on Red Hat OpenShift Container Platform, you can use the OpenShift Web console or the **oc** command.

For instructions about using the Web console, see Create and build an image using the Web console .

For detailed instructions about using the **oc** command, see CLI Reference. The following commands are likely to be required:

- To create a project, use the following command:

```
$ oc new-project <project-name>
```

  For more information, see Creating a project using the CLI .

- To deploy a template (create an application from a template), use the following command:

```
$ oc new-app -f <template-name> -p <parameter>=<value> -p <parameter>=<value> ...
```

  For more information, see Creating an application using the CLI .

- To view a list of the active pods in the project, use the following command:

```
$ oc get pods
```

- To view the current status of a pod, including information whether or not the pod deployment has completed and it is now in a running state, use the following command:

```
$ oc describe pod <pod-name>
```

  You can also use the **oc describe** command to view the current status of other objects. For more information, see Application modification operations.

- To view the logs for a pod, use the following command:

```
$ oc logs <pod-name>
```

- To view deployment logs, look up a **DeploymentConfig** name in the template reference and enter the following command:

```
$ oc logs -f dc/<deployment-config-name>
```

  For more information, see Viewing deployment logs.

- To view build logs, look up a **BuildConfig** name in the template reference and enter the command:

```
$ oc logs -f bc/<build-config-name>
```

  For more information, see Accessing build logs.

- To scale a pod in the application, look up a **DeploymentConfig** name in the template reference and enter the command:

```
$ oc scale dc/<deployment-config-name> --replicas=<number>
```

For more information, see Manual scaling.

- To undeploy the application, you can delete the project by using the command:

```
$ oc delete project <project-name>
```

Alternatively, you can use the **oc delete** command to remove any part of the application, such as a pod or replication controller. For details, see Application modification operations.

APPENDIX A. VERSIONING INFORMATION

Documentation last updated on Friday, June 25, 2021.