Red Hat Decision Manager 7.8

Deploying a Red Hat Decision Manager authoring or managed server environment on Red Hat OpenShift Container Platform

Last Updated: 2020-09-08

Red Hat Decision Manager 7.8 Deploying a Red Hat Decision Manager authoring or managed server 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 Decision Manager 7.8 authoring or managed server environment on Red Hat OpenShift Container Platform.

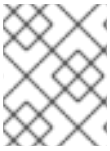
Table of Contents

PREFACE

As a system engineer, you can deploy a Red Hat Decision Manager authoring or managed environment on Red Hat OpenShift Container Platform to provide a platform for developing or running services 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.
- For a high-availability deployment, the following resources are available on the OpenShift cluster:
  - For the Business Central replicated pod, 8 gigabytes of memory and 2 CPU cores are required for each replica. Two replicas are created by default.
  - For the KIE Server replicated pod, 1 gigabyte of memory and 1 CPU core are required for each replica. Two replicas are created by default.
  - The Red Hat AMQ replicated pod uses the default resource limits configured on your cluster.
  - For the Red Hat Data Grid replicated pod, 2 gigabytes of memory and 1 CPU core are required for each replica. Two replicas are created by default.



NOTE

For instructions about checking the capacity of your cluster, see Analyzing cluster capacity in the Red Hat OpenShift Container Platform 3.11 product documentation.

- The OpenShift project for the deployment is 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, a sufficient persistent volume must be available. By default, Business Central requires one 1Gi PV. You can change the PV size for Business Central persistent storage in the template parameters.
- If you intend to deploy a high-availability authoring environment, which includes high-availability Business Central, your OpenShift environment supports persistent volumes with **ReadWriteMany** mode. If your environment does not support this mode, you can use NFS to provision the volumes. However, for best performance and reliability, use GlusterFS to provision persistent volumes for a high-availability authoring environment. For information about access mode support in OpenShift public and dedicated clouds, see Access Modes.

NOTE

Since Red Hat Decision Manager version 7.5, images and templates for Red Hat OpenShift Container Platform 3.x are deprecated. These images and templates do not get new features, but remain supported until the end of full support for Red Hat OpenShift Container Platform version 3.x. For more information about the full support lifecycle phase for Red Hat OpenShift Container Platform version 3.x, see Red Hat OpenShift Container Platform Life Cycle Policy (non-current versions).

NOTE

Do not use Red Hat Decision Manager templates with Red Hat OpenShift Container Platform 4.x. To deploy Red Hat Decision Manager on Red Hat OpenShift Container Platform 4.x, see the instructions in *Deploying a Red Hat Decision Manager environment on Red Hat OpenShift Container Platform using Operators*.

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

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

In this solution, components of Red Hat Decision Manager are deployed as separate OpenShift pods. You can scale each of the pods up and down individually 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 Decision Manager are available on OpenShift:

- KIE Server, also known as *Execution Server*, is the infrastructure element that runs decision services and other deployable assets (collectively referred to as *services*) . All logic of the services runs on execution servers.
  In some templates, you can scale up a KIE 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 run the same services. OpenShift provides load balancing and a request can be handled by any of the pods.

  You can deploy a separate KIE 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 KIE Server pods as required.

- Business Central is a web-based interactive environment used for authoring services. It also provides a management console. You can use Business Central to develop services and deploy them to KIE Servers.
  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.

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

The following environment types are typical:

- *Authoring or managed environment* : An environment architecture that can be used for creating and modifying services using Business Central and also for running services on KIE Servers. It consists of pods that provide Business Central for the authoring work and one or more KIE Servers for execution of the services. Each KIE Server is a pod that you can replicate by scaling it up or down as necessary. You can deploy and undeploy services on each KIE Server using Business Central. For instructions about deploying this environment, see *Deploying a Red Hat Decision Manager authoring or 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 KIE 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 KIE 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. For instructions about deploying this environment, see *Deploying a Red Hat Decision 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 KIE 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 Decision Manager trial environment on Red Hat OpenShift Container Platform*.

To deploy a Red Hat Decision Manager environment on OpenShift, you can use the templates that are provided with Red Hat Decision Manager.

CHAPTER 2. ARCHITECTURE OF AN AUTHORING ENVIRONMENT

In Red Hat Decision Manager, the Business Central component provides a web-based interactive user interface for authoring services. The KIE Server component runs the services.

You can also use Business Central to deploy services onto a KIE Server. You can use several KIE Servers to run different services and control the servers from the same Business Central.

Single authoring environment

In a single authoring environment, only one instance of Business Central is running. Multiple users can access its web interface at the same time, however the performance can be limited and there is no failover capability.

Business Central includes a built-in Maven repository that stores the built versions of the services that you develop (KJAR files/artifacts). You can use your continuous integration and continuous deployment (CICD) tools to retrieve these artifacts from the repository and move them as necessary.

Business Central saves the source code in a built-in Git repository, stored in the **.niogit** directory. It uses a built-in indexing mechanism to index the assets in your services.

Business Central uses persistent storage for the Maven repository and for the Git repository.

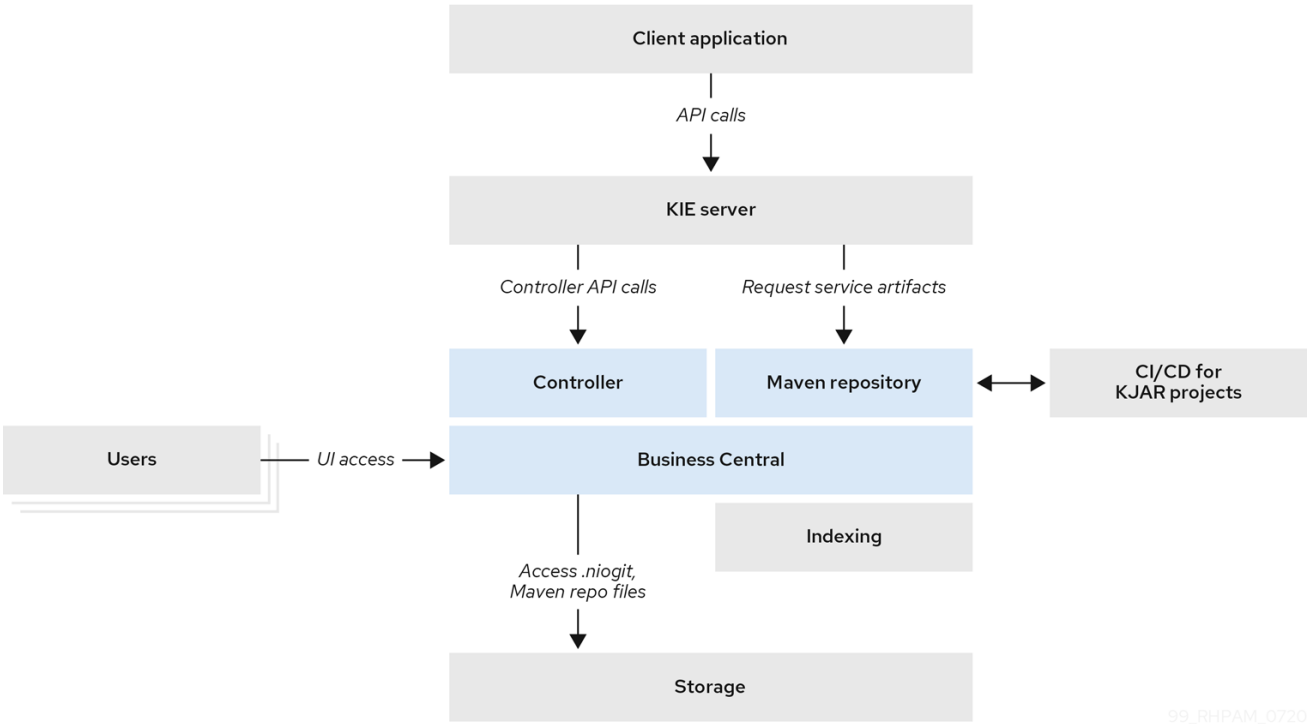
A single authoring environment, by default, includes one KIE Server.

A single authoring environment, by default, uses the *controller strategy*. Business Central includes the *Controller*, a component that can manage KIE Servers. When you configure a KIE Server to connect to Business Central, the KIE Server uses a REST API to connect to the Controller. This connection opens a persistent WebSocket. In an OpenShift deployment that uses the controller strategy, each KIE Server is initially configured to connect to the Business Central Controller.

When you use the Business Central user interface to deploy or manage a service on the KIE Server, the KIE Server receives the request through the Controller connection WebSocket. To deploy a service, the KIE Server requests the necessary artifact from the Maven repository that is a part of Business Central.

Client applications use a REST API to use services that run on the KIE Server.

Figure 2.1. Architecture diagram for a single authoring environment



Clustering KIE Servers and using multiple KIE Servers

You can scale a KIE Server pod to run a clustered KIE Server environment.

In a clustered deployment, several instances of the KIE Server run the same services. These servers can connect to the Business Central Controller using the same server ID, so they can receive the same requests from the controller. Red Hat OpenShift Container Platform provides load-balancing between the servers. The services that run on a clustered KIE Server must be stateless, because requests from the same client might be processed by different instances.

You can also deploy several independent KIE Servers to run different services. In this case, the servers connect to the Business Central Controller with different server ID values. You can use the Business Central UI to deploy services to each of the servers.

Smart Router

The optional Smart Router component provides a layer between client applications and KIE Servers. It can be useful if you are using several independent KIE Servers.

The client application can use services running on different KIE Servers, but always connects to the Smart Router. The Smart Router automatically passes the request to the KIE Servers that runs the required service. The Smart Router also enables management of service versions and provides an additional load-balancing layer.

High-availability authoring environment

In a high-availability (HA) authoring environment, the Business Central pod is scaled, so several instances of Business Central are running. Red Hat OpenShift Container Platform provides load balancing for user requests. This environment provides optimal performance for multiple users and supports failover.

Each instance of Business Central includes the Maven repository for the built artifacts and uses the **.niogit** Git repository for source code. The instances use shared persistent storage for the repositories. A persistent volume with **ReadWriteMany** access is required for this storage.

An instance of Red Hat DataGrid provides indexing of all projects and assets developed in Business Central.

An instance of Red Hat AMQ propagates Java CDI messages between all instances of Business Central. For example, when a new project is created or when an asset is locked or modified on one of the instances, this information is immediately reflected in all other instances.

The controller strategy is not suitable for clustered deployment. In an OpenShift deployment, a high-availability Business Central must manage KIE Servers using the *OpenShift startup strategy*.

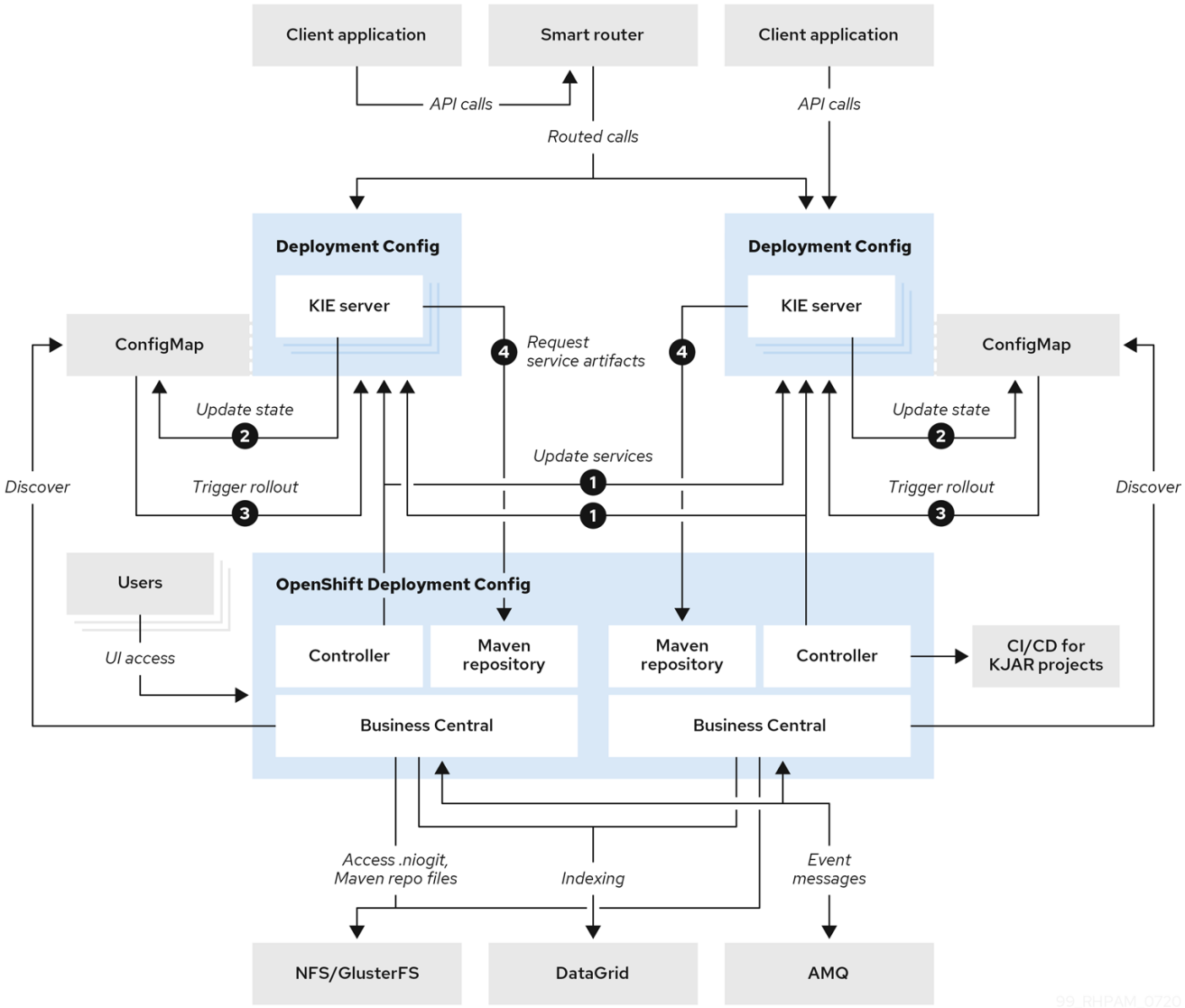
Each KIE Server deployment (which can be scaled) creates a ConfigMap that reflects its current state. The Business Central discovers all KIE Servers by reading their ConfigMaps.

When the user requests a change in KIE Server configuration (for example, deploys or undeploys a service), Business Central initiates a connection to the KIE Server and sends a REST API request. The KIE Server changes the ConfigMap to reflect the new configuration state and then triggers its own redeployment, so that all instances are redeployed and reflect the new configuration.

You can deploy several independent KIE Servers in your OpenShift environment. Each of the KIE Servers has a separate ConfigMap with the necessary configuration. You can scale each of the KIE Servers separately.

You can include Smart Router in the OpenShift deployment.

Figure 2.2. Architecture diagram for a high-availability authoring environment



CHAPTER 3. PREPARING TO DEPLOY RED HAT DECISION MANAGER IN YOUR OPENSHIFT ENVIRONMENT

Before deploying Red Hat Decision 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 decision services or for other decision services

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

To deploy Red Hat Decision 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 rhdm78-decisioncentral-openshift
$ oc get imagestreamtag -n openshift | grep rhdm78-kieserver-openshift
```

   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 **rhdm-7.8.0-openshift-templates.zip** product deliverable file from the Software Downloads page and extract the **rhdm78-image-streams.yaml** file.

g. Enter the following command:

```
$ oc apply -f rhdm78-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.

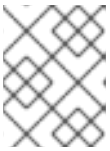
3.2. CREATING THE SECRETS FOR KIE SERVER

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 Red Hat OpenShift Container Platform documentation.

You must create an SSL certificate for HTTP access to KIE 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 KIE 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 KIE 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 Decision Manager configuration is **jboss**.

4. Record the password of the keystore file. The default value for this name in Red Hat Decision 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
```

3.3. 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 KIE Server.

Procedure

1. Generate an SSL keystore with a private and public key for SSL encryption for KIE 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 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 Decision Manager configuration is **jboss**.
4. Record the password of the keystore file. The default value for this name in Red Hat Decision Manager configuration is **mykeystorepass**.
5. Use the **oc** command to generate a secret named **decisioncentral-app-secret** from the new keystore file:

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

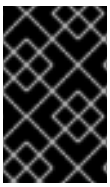
3.4. CREATING THE SECRET FOR THE ADMINISTRATIVE USER

You must create a generic secret that contains the user name and password for a Red Hat Decision Manager administrative user account. This secret is required for deploying Red Hat Decision Manager using any template except the trial template.

The secret must contain the user name and password as literals. The key name for the user name is **KIE_ADMIN_USER**. The key name for the password is **KIE_ADMIN_PWD**.

If you are using multiple templates to deploy components of Red Hat Decision Manager, use the same secret for all these deployments. The components utilize this user account to communicate with each other.

You can also use this user account to log in to Business Central.



IMPORTANT

If you use RH-SSO or LDAP authentication, the same user with the same password must be configured in your authentication system with the **kie-server,rest-all,admin** roles for Red Hat Decision Manager.

Procedure

Use the **oc** command to generate a generic secret named **kie-admin-user-secret** from the user name and password:

```
$ oc create secret generic rhpam-credentials --from-literal=KIE_ADMIN_USER=adminUser --from-literal=KIE_ADMIN_PWD=adminPassword
```

In this command, replace *adminPassword* with the password for the administrative user. Optionally, you can replace *adminUser* with another user name for the administrative user.

3.5. 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. Configure a Maven release repository to which you have write access. The repository must allow read access without authentication and your OpenShift environment must have network 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 in the Red Hat OpenShift Container Platform 3.11 documentation. 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. Click **Red Hat Process Automation Manager 7.8.0 Offliner Content List** to download the **rhdm-7.8.0-offliner.zip** product deliverable file from the Software Downloads page of the Red Hat Customer Portal.

   b. Extract the contents of the **rhdm-7.8.0-offliner.zip** file into any directory.

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

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

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

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

d. Upload all artifacts from the **repository** subdirectory to the Maven mirror repository that you prepared. You can use the Maven Repository Provisioner utility, available from the Maven repository tools Git repository, 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, available from the Maven repository tools Git repository, to upload the artifacts.

3.6. 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
```

3.7. PROVISIONING PERSISTENT VOLUMES WITH READWRITEMANY ACCESS MODE USING NFS

If you want to deploy high-availability Business Central, your environment must provision persistent volumes with **ReadWriteMany** access mode.

NOTE

If you want to deploy a high-availability authoring environment, for optimal performance and reliability, provision persistent volumes using GlusterFS. Configure the GlusterFS storage class as described in Section 3.6, “Changing GlusterFS configuration”.

If your configuration requires provisioning persistent volumes with **ReadWriteMany** access mode but your environment does not support such provisioning, use NFS to provision the volumes. Otherwise, skip this procedure.

Procedure

Deploy an NFS server and provision the persistent volumes using NFS. For information about provisioning persistent volumes using NFS, see the "Persistent storage using NFS" section of the *Configuring Clusters* guide in the Red Hat OpenShift Container Platform 3.11 documentation.

CHAPTER 4. AUTHORING OR MANAGED SERVER ENVIRONMENT

You can deploy an environment for creating and modifying services using Business Central and for running them in KIE Servers managed by Business Central. This environment consists of Business Central and one or more KIE Servers.

You can use Business Central both to develop services and to deploy them to KIE Servers. You can connect several KIE Servers to one Business Central to manage deployment of services to each of the servers.

If necessary, you can create separate environments, so that you can use one deployment of Business Central to author services (*authoring environment*) and another deployment of Business Central to manage deployment of staging or production services on several KIE Servers (*managed server environment*). Usually, one KIE Server is sufficient for a dedicated authoring environment. You can use an external Maven repository to store services from an authoring environment and deploy them to a separate managed server environment.

For Red Hat Decision Manager, the procedures to deploy an authoring environment and a managed server environment are the same. You must first deploy an authoring environment template, consisting of Business Central and one KIE Server.

If necessary, you can deploy additional KIE Server templates in the same namespace to create an environment with multiple KIE Servers. This environment can be a managed server environment for staging and production deployment of services.

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

A single authoring environment contains two pods. One of the pods runs Business Central, the other runs KIE Server. This environment is most suitable for single-user authoring or when your OpenShift infrastructure has limited resources. It does not require persistent volumes that support the **ReadWriteMany** access mode.

In a single authoring environment, you cannot scale Business Central. You can scale KIE Server.

In an HA authoring environment, both Business Central and KIE Server are provided in scalable pods. When pods are scaled, persistent storage is shared between the copies.

To enable high-availability functionality in Business Central, additional pods with AMQ and Data Grid are required. These pods are configured and deployed by the high-availability authoring template. Use a high-availability authoring environment to provide maximum reliability and responsiveness, especially if several users are involved in authoring at the same time.

In the current version of Red Hat Decision Manager, an HA authoring environment is supported with certain limitations:

- If a Business Central pod crashes while a user works with it, the user can get an error message and then is redirected to another pod. Logging on again is not required.
- If a Business Central pod crashes during a user operation, data that was not committed (saved) might be lost.
- If a Business Central pod crashes during creation of a project, an unusable project might be created.

- If a Business Central pod crashes during creation of an asset, the asset might be created but not indexed, so it cannot be used. The user can open the asset in Business Central and save it again to make it indexed.

- When a user deploys a service to the KIE Server, the KIE Server deployment is rolled out again. Users can not deploy another service to the same KIE Server until the roll-out completes.

In a high-availability authoring environment you can also deploy additional managed or immutable KIE Servers, if required. Business Central can automatically discover any KIE Servers in the same namespace, including immutable KIE Servers and managed KIE Servers.

If you want to deploy additional managed or immutable KIE Servers in a single authoring environment, you must complete an additional manual step to enable the **OpenShiftStartupStrategy** setting in the environment, as described in Section 4.5, "Enabling the **OpenShiftStartupStrategy** setting to connect additional KIE Servers to Business Central". This setting enables the discovery of other KIE Servers.

For instructions about deploying managed KIE Servers, see Section 4.6, "Deploying an additional managed KIE Server for an authoring or managed environment". For instructions about deploying immutable KIE Servers, see *Deploying a Red Hat Decision Manager immutable server environment on Red Hat OpenShift Container Platform*.

4.1. 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 KIE Server.

4.1.1. Starting configuration of the template for an authoring environment

If you want to deploy a single authoring environment, use the **rhdm78-authoring.yaml** template file.

If you want to deploy a high-availability authoring environment, use the **rhdm78-authoring-ha.yaml** template file.

Procedure

1. Download the **rhdm-7.8.0-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
     DECISION_CENTRAL_HTTPS_SECRET=decisioncentral-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 4.1.2, "Setting required parameters for an authoring environment" to set common parameters. You can view the template file to see descriptions for all parameters.

4.1.2. 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 4.1.1, "Starting configuration of the template for an authoring environment".

Procedure

1. Set the following parameters:
   - **Credentials secret** (**CREDENTIALS_SECRET**): The name of the secret containing the administrative user credentials, as created in Section 3.4, "Creating the secret for the administrative user".
   - **Business Central Server Keystore Secret Name** (**DECISION_CENTRAL_HTTPS_SECRET**): The name of the secret for Business Central, as created in Section 3.3, "Creating the secrets for Business Central" .
   - **KIE Server Keystore Secret Name**(**KIE_SERVER_HTTPS_SECRET**): The name of the secret for KIE Server, as created in Section 3.2, "Creating the secrets for KIE Server".
   - **Business Central Server Certificate Name**(**DECISION_CENTRAL_HTTPS_NAME**): The name of the certificate in the keystore that you created in Section 3.3, "Creating the secrets for Business Central".
   - **Business Central Server Keystore Password** (**DECISION_CENTRAL_HTTPS_PASSWORD**): The password for the keystore that you created in Section 3.3, "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 3.2, "Creating the secrets for KIE Server".
   - **KIE Server Keystore Password** (**KIE_SERVER_HTTPS_PASSWORD**): The password for the keystore that you created in Section 3.2, "Creating the secrets for KIE Server".
   - **Application Name** (**APPLICATION_NAME**): The name of the OpenShift application. It is used in the default URLs for Business Central Monitoring and KIE Server. OpenShift uses the application name to create a separate set of deployment configurations, services, routes, labels, and artifacts.
   - **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 3.1, "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.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 4.1.12, "Completing deployment of the template for an authoring environment".

4.1.3. 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 4.1.1, "Starting configuration of the template for an authoring environment".

Procedure

If you installed an image streams file according to instructions in Section 3.1, "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.

4.1.4. 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 4.1.1, "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 user name 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 4.1.12, "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 Decision Manager project* .

4.1.5. 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 3.5, "Preparing a Maven mirror repository for offline use" .

Prerequisites

- You started the configuration of the template, as described in Section 4.1.1, "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 3.5, "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-rhdmcentr**; 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 4.1.12, "Completing deployment of the template for an authoring environment".

4.1.6. Configuring Business Central and KIE Server replicas for a high-availability authoring environment

If you are deploying a high-availability authoring environment, by default two replicas of Business Central and two replicas of the KIE Server are initially created.

Optionally, you can modify the number of replicas.

Skip this procedure for a single authoring environment.

Prerequisites

- You started the configuration of the template, as described in Section 4.1.1, "Starting configuration of the template for an authoring environment".

Procedure

To modify the numbers of initial replicas, set the following parameters:

- **Business Central Container Replicas**(**DECISION_CENTRAL_CONTAINER_REPLICAS**): The number of replicas that the deployment initially creates for Business Central.
- **KIE Server Container Replicas**(**KIE_SERVER_CONTAINER_REPLICAS**): The number of replicas that the deployment initially creates for the KIE Server.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 4.1.12, "Completing deployment of the template for an authoring environment".

4.1.7. 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 4.1.1, "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 4.2, "(Optional) Providing the Git hooks directory".

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 4.1.12, "Completing deployment of the template for an authoring environment".

4.1.8. Configuring resource usage for a high-availability deployment

If you are deploying the high-availability template (**rhdm78-authoring-ha.yaml**), you can optionally configure resource usage to optimize performance for your requirements.

If you are deploying the single authoring environment template (**rhdm78-authoring.yaml**), skip this procedure.

For more information about sizing resources, see the following sections in the Red Hat OpenShift Container Platform 3.11 product documentation:

- Application memory sizing
- Compute resources

Prerequisites

- You started the configuration of the template, as described in Section 4.1.1, "Starting configuration of the template for an authoring environment".

Procedure

Set the following parameters of the template as applicable:

- **Business Central Container Memory Limit**(**DECISION_CENTRAL_MEMORY_LIMIT**): The amount of memory requested in the OpenShift environment for the Business Central container. The default value is **8Gi**.
- **Business Central JVM Max Memory Ratio** (**DECISION_CENTRAL_JAVA_MAX_MEM_RATIO**): The percentage of container memory that is used for the Java Virtual Machine for Business Central. The remaining memory is used for the operating system. The default value is **80**, for a limit of 80%.
- **Business Central Container CPU Limit**(**DECISION_CENTRAL_CPU_LIMIT**): The maximum CPU usage for Business Central. The default value is **2000m**.
- **KIE Server Container Memory Limit**(**KIE_SERVER_MEMORY_LIMIT**): The amount of memory requested in the OpenShift environment for the KIE Server container. The default value is **1Gi**.
- **KIE Server Container CPU Limit**(**KIE_SERVER_CPU_LIMIT**): The maximum CPU usage for KIE Server. The default value is **1000m**.
- **DataGrid Container Memory Limit**(**DATAGRID_MEMORY_LIMIT**): The amount of memory requested in the OpenShift environment for the Red Hat Data Grid container. The default value is **2Gi**.
- **DataGrid Container CPU Limit**(**DATAGRID_CPU_LIMIT**): The maximum CPU usage for Red Hat Data Grid. The default value is **1000m**.

4.1.9. 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 Decision Manager is created in the RH-SSO authentication system.

- User names and passwords for Red Hat Decision Manager are created in the RH-SSO authentication system. For a list of the available roles, see Chapter 5, *Red Hat Decision Manager roles and users*.
  You must create a user with the username and password configured in the secret for the administrative user, as described in Section 3.4, "Creating the secret for the administrative user". This user must have the **kie-server,rest-all,admin** roles.

- Clients are created in the RH-SSO authentication system for all components of the Red Hat Decision 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 Decision 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 4.1.1, "Starting configuration of the template for an authoring environment".

Procedure

1. 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 Decision 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.
2. Complete one of the following procedures:
   a. If you created the clients for Red Hat Decision Manager within RH-SSO, set the following parameters in the template:
      - **Business Central RH-SSO Client name** (**DECISION_CENTRAL_SSO_CLIENT**): The RH-SSO client name for Business Central.
      - **Business Central RH-SSO Client Secret** (**DECISION_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 KIE Server.
      - **KIE Server RH-SSO Client Secret** (**KIE_SERVER_SSO_SECRET**): The secret string that is set in RH-SSO for the client for KIE Server.

   b. To create the clients for Red Hat Decision Manager within RH-SSO, set the following parameters in the template:
      - **Business Central RH-SSO Client name** (**DECISION_CENTRAL_SSO_CLIENT**): The name of the client to create in RH-SSO for Business Central.
      - **Business Central RH-SSO Client Secret** (**DECISION_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 KIE Server.

- **KIE Server RH-SSO Client Secret**(**KIE_SERVER_SSO_SECRET**): The secret string to set in RH-SSO for the client for KIE 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 Decision Manager. You must provide this user name and password in order to create the required clients.

Next steps

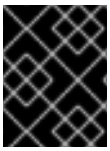
If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 4.1.12, "Completing deployment of the template for an authoring environment".

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

4.1.10. 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 Decision Manager in the LDAP system. For a list of the available roles, see Chapter 5, *Red Hat Decision Manager roles and users* .
  You must create a user with the username and password configured in the secret for the administrative user, as described in Section 3.4, "Creating the secret for the administrative user". This user must have the **kie-server,rest-all,admin** roles.
- You started the configuration of the template, as described in Section 4.1.1, "Starting configuration of the template for an authoring environment".

Procedure

1. 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 Decision 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 4.4, "(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 4.1.12, "Completing deployment of the template for an authoring environment".

4.1.11. Enabling Prometheus metric collection for an authoring environment

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

Prerequisites

- You started the configuration of the template, as described in Section 4.1.1, "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 4.1.12, "Completing deployment of the template for an authoring environment".

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

4.1.12. 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.

4.2. (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>-rhdmcentr --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
```

NOTE

A **pre-commit** script is not supported in Business Central. Use a **post-commit** script.

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>-rhdmcentr --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*-rhdmcentr** 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 KIE 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 KIE 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.

4.3. (OPTIONAL) PROVIDING A TRUSTSTORE FOR ACCESSING HTTPS SERVERS WITH SELF-SIGNED CERTIFICATES

Components of your Red Hat Decision Manager infrastructure might need to use HTTPS access to servers that have a self-signed HTTPS certificate. For example, Business Central and KIE Server might need to interact with an internal Nexus repository that uses a self-signed HTTPS server certificate.

In this case, to ensure that HTTPS connections complete successfully, you must provide client certificates for these services using a truststore.

Skip this procedure if you do not need Red Hat Decision Manager components to communicate with servers that use self-signed HTTPS server certificates.

Procedure

1. Prepare a truststore with the certificates. Use the following command to create a truststore or to add a certificate to an existing truststore. Add all the necessary certificates to one truststore.

```
keytool -importcert -file certificate-file -alias alias -keyalg algorithm -keysize size -trustcacerts -noprompt -storetype JKS -keypass truststore-password -storepass truststore-password -keystore keystore-file
```

Replace the following values:

- ***certificate-file***: The pathname of the certificate that you want to add to the truststore.
- ***alias***: The alias for the certificate in the truststore. If you are adding more than one certificate to the truststore, every certificate must have a unique alias.
- ***algorithm***: The encryption algorithm used for the certificate, typically **RSA**.
- ***size***: The size of the certificate key in bytes, for example, **2048**.
- ***truststore-password***: The password for the truststore.
- ***keystore-file***: The pathname of the truststore file. If the file does not exist, the command creates a new truststore.
  The following example command adds a certificate from the **/var/certs/nexus.cer** file to a truststore in the **/var/keystores/custom-trustore.jks** file. The truststore password is **mykeystorepass**.

```
keytool -importcert -file /var/certs/nexus.cer -alias nexus-cert -keyalg RSA -keysize 2048
-trustcacerts -noprompt -storetype JKS -keypass mykeystorepass -storepass
mykeystorepass -keystore /var/keystores/custom-trustore.jks
```

2. Create a secret with the truststore file using the **oc** command, for example:

```
oc create secret generic truststore-secret --from-file=/var/keystores/custom-trustore.jks
```

3. In the deployment for the necessary components of your infrastructure, mount the secret and then set the **JAVA_OPTS_APPEND** option to enable the Java application infrastructure to use the trast store, for example:

```
oc set volume dc/myapp-rhdmcentr --add --overwrite --name=custom-trustore-volume --
mount-path /etc/custom-secret-volume --secret-name=custom-secret

oc set env dc/myapp-rhdmcentr JAVA_OPTS_APPEND='-
Djavax.net.ssl.trustStore=/etc/custom-secret-volume/custom-trustore.jks -
Djavax.net.ssl.trustStoreType=jks -Djavax.net.ssl.trustStorePassword=mykeystorepass'
```

```
oc set volume dc/myapp-kieserver --add --overwrite --name=custom-trustore-volume --
mount-path /etc/custom-secret-volume --secret-name=custom-secret

oc set env dc/myapp-kieserver JAVA_OPTS_APPEND='-
Djavax.net.ssl.trustStore=/etc/custom-secret-volume/custom-trustore.jks -
Djavax.net.ssl.trustStoreType=jks -Djavax.net.ssl.trustStorePassword=mykeystorepass'
```

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

4.4. (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*-rhdmcentr**: Business Central
- ***myapp*-kieserver**: KIE Server

Replace **myapp** with the application name. Sometimes, several KIE 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** .

4.5. ENABLING THE OPENSHIFTSTARTUPSTRATEGY SETTING TO CONNECT ADDITIONAL KIE SERVERS TO BUSINESS CENTRAL

In an environment deployed using Red Hat Decision Manager authoring templates, Business Central manages one KIE Server. You can scale the KIE Server pod, but all the copies execute the same services.

You can connect additional KIE Servers to Business Central. However, if you deployed a single authoring environment using the **rhdm78-authoring.yaml**, you must enable the **OpenShiftStartupStrategy** setting in the environment. When **OpenShiftStartupStrategy** is enabled, Business Central automatically discovers KIE Servers in the same namespace and these KIE Servers can be configured to connect to the Business Central.

With the **OpenShiftStartupStrategy** setting, when a user deploys a service to the KIE Server, the KIE Server deployment is rolled out again. Users can not deploy another service to the same KIE Server until

the roll-out completes. Because the roll-out might take noticeable time, the **OpenShiftStartupStrategy** setting might not be suitable for some authoring environments.

Do not complete this procedure if you deployed a high-availability authoring environment using the **rhdm78-authoring-ha.yaml** template. In this environment, the **OpenShiftStartupStrategy** setting is enabled by default.

Do not complete this procedure unless you want to connect additional KIE Servers to Business Central.

Prerequisites

- You deployed an authoring environment using the **rhdm78-authoring.yaml** template.
- You are logged in to the OpenShift project where the environment is deployed using the **oc** tool.

Procedure

1. Enter the following command to view the deployment configurations that are deployed in the project:

```
$ oc get dc
```

2. In the output of the command, find the deployment configuration names for the Business Central and KIE Server pods:
   - The name of the deployment configuration for Business Central is ***myapp*-rhdmcentr**. Replace ***myapp*** with the application name of the environment, which is set in the **APPLICATION_NAME** parameter of the template.
   - The name of the deployment configuration for KIE Server is ***myapp*-kieserver**. Replace ***myapp*** with the application name.
3. Enter the following commands to enable the **OpenShiftStartupStrategy** setting on the pods:

```
$ oc env myapp-rhdmcentr KIE_SERVER_CONTROLLER_OPENSHIFT_ENABLED=true
$ oc env myapp-kieserver KIE_SERVER_STARTUP_STRATEGY=OpenShiftStartupStrategy
```

   In these commands, replace ***myapp*-rhdmcentr** with the Business Central deployment configuration name and ***myapp*-kieserver** with the KIE Server deployment configuration name.
4. When you enable the **OpenShiftStartupStrategy** setting, by default Business Central discovers only KIE Servers that are deployed with the same value of the **APPLICATION_NAME** parameter as the authoring template. If you want to connect KIE Servers with any other application names to the Business Central, enter the following command:

```
$ oc env myapp-rhdmcentr
KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED=true
```

   In this command, replace ***myapp*-rhdmcentr** with the Business Central deployment configuration name.

4.6. DEPLOYING AN ADDITIONAL MANAGED KIE SERVER FOR AN AUTHORING OR MANAGED ENVIRONMENT

You can deploy an additional managed KIE Server to an authoring or managed environment. Deploy the server in the same project as the Business Central deployment.

If you deployed a single authoring environment using the **rhdm78-authoring.yaml** template, you must enable the **OpenShiftStartupStrategy** setting for your environment for the Business Central to connect to the KIE Server. For instructions about enabling the **OpenShiftStartupStrategy** setting, see Section 4.5, "Enabling the **OpenShiftStartupStrategy** setting to connect additional KIE Servers to Business Central". You do not need to complete this procedure for a high-availability authoring environment.

The KIE Server loads services from a Maven repository. You must configure the server to use either the Business Central built-in repository or an external repository.

The server starts with no loaded services. Use Business Central or the REST API of the KIE Server to deploy and undeploy services on the server.

4.6.1. Starting configuration of the template for an additional managed KIE Server

To deploy an additional managed KIE Server, use the **rhdm78-kieserver.yaml** template file.

Procedure

1. Download the **rhdm-7.8.0-openshift-templates.zip** product deliverable file from the Software Downloads page of the Red Hat Customer Portal.
2. Extract the **rhdm78-kieserver.yaml** 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 **rhdm78-kieserver.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>/rhdm78-kieserver.yaml -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.
     - 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 4.6.2, "Setting required parameters for an additional managed KIE Server" to set common parameters. You can view the template file to see descriptions for all parameters.

4.6.2. Setting required parameters for an additional managed KIE Server

When configuring the template to deploy an additional managed KIE Server, you must set the following parameters in all cases.

Prerequisites

- You started the configuration of the template, as described in Section 4.6.1, "Starting configuration of the template for an additional managed KIE Server".

Procedure

1. Set the following parameters:

   - **Credentials secret** (**CREDENTIALS_SECRET**): The name of the secret containing the administrative user credentials, as created in Section 3.4, "Creating the secret for the administrative user".
   - **KIE Server Keystore Secret Name**(**KIE_SERVER_HTTPS_SECRET**): The name of the secret for KIE Server, as created in Section 3.2, "Creating the secrets for KIE Server".
   - **KIE Server Certificate Name**(**KIE_SERVER_HTTPS_NAME**): The name of the certificate in the keystore that you created in Section 3.2, "Creating the secrets for KIE Server".
   - **KIE Server Keystore Password** (**KIE_SERVER_HTTPS_PASSWORD**): The password for the keystore that you created in Section 3.2, "Creating the secrets for KIE Server".
   - **Application Name** (**APPLICATION_NAME**): The name of the OpenShift application. It is used in the default URLs for Business Central Monitoring and KIE Server. OpenShift uses the application name to create a separate set of deployment configurations, services, routes, labels, and artifacts. You can deploy several applications using the same template into the same project, as long as you use different application names. Also, the application name determines the name of the server configuration (server template) that the KIE Server joins on Business Central. If you are deploying several KIE Servers, you must ensure each of the servers has a different application name.
   - **KIE Server Mode**(**KIE_SERVER_MODE**): In the **rhdm78-kieserver.yaml** template the default value is **PRODUCTION**. In **PRODUCTION** mode, you cannot deploy **SNAPSHOT** versions of KJAR artifacts on the KIE Server and cannot change versions of an artifact in an existing container. To deploy a new version with **PRODUCTION** mode, create a new container on the same KIE Server. To deploy **SNAPSHOT** versions or to change versions of an artifact in an existing container, set this parameter to **DEVELOPMENT**.
   - **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 3.1, "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.

Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 4.6.9, "Completing deployment of the template for an additional managed KIE Server".

4.6.3. Configuring the image stream namespace for an additional managed KIE Server

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 4.6.1, "Starting configuration of the template for an additional managed KIE Server".

Procedure

If you installed an image streams file according to instructions in Section 3.1, "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.

4.6.4. Configuring information about a Business Central instance for an additional managed KIE Server

If you want to enable a connection from a Business Central instance in the same namespace to the KIE Server, you must configure information about the Business Central instance.

The Business Central instance must be configured with the same credentials secret (**CREDENTIALS_SECRET**) as the KIE Server.

Prerequisites

- You started the configuration of the template, as described in Section 4.6.1, "Starting configuration of the template for an additional managed KIE Server".

Procedure

1. Set the following parameters:
   - **Name of the Business Central service**(**DECISION_CENTRAL_SERVICE**): The OpenShift service name for the Business Central.
2. Configure access to the Maven repository from which the server must load services. You must configure the same repository that the Business Central uses.
   - If the Business Central uses its own built-in repository, set the following parameter:
     - **Name of the Maven service hosted by Business Central** (**DECISION_CENTRAL_MAVEN_SERVICE**): The OpenShift service name for the Business Central.
   - If you configured the Business Central to use an external Maven repository, set the following parameters:
     - **Maven repository URL** (**MAVEN_REPO_URL**): A URL for the external Maven repository that Business Central uses.
     - **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 user name 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 4.6.9, "Completing deployment of the template for an additional managed KIE Server".

4.6.5. Configuring access to a Maven mirror in an environment without a connection to the public Internet for an additional managed KIE Server

When configuring the template to deploy an additional managed KIE Server, 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 3.5, "Preparing a Maven mirror repository for offline use" .

Prerequisites

- You started the configuration of the template, as described in Section 4.6.1, "Starting configuration of the template for an additional managed KIE Server".

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 3.5, "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:***. With this value, Maven retrieves every required artifact from the mirror and does not query any other repositories.
  - If you configure an external Maven repository (**MAVEN_REPO_URL**), change **MAVEN_MIRROR_OF** to exclude the artifacts in this repository from the mirror, for example, **external:*,!repo-custom**. Replace **repo-custom** with the ID that you configured in **MAVEN_REPO_ID**.
  - If you configure a built-in Business Central Maven repository (**DECISION_CENTRAL_MAVEN_SERVICE**), change **MAVEN_MIRROR_OF** to exclude the artifacts in this repository from the mirror: **external:*,!repo-rhdmcentr**.
  - If you configure both repositories, change **MAVEN_MIRROR_OF** to exclude the artifacts in both repositories from the mirror: **external:*,!repo-rhdmcentr,!repo-custom**. Replace **repo-custom** with the ID that you configured in **MAVEN_REPO_ID**.

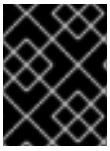
Next steps

If necessary, set additional parameters.

To complete the deployment, follow the procedure in Section 4.6.9, "Completing deployment of the template for an additional managed KIE Server".

4.6.6. Setting parameters for RH-SSO authentication for an additional managed KIE Server

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



IMPORTANT

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

Prerequisites

- A realm for Red Hat Decision Manager is created in the RH-SSO authentication system.
- User names and passwords for Red Hat Decision Manager are created in the RH-SSO authentication system. For a list of the available roles, see Chapter 5, *Red Hat Decision Manager roles and users*.
  You must create a user with the username and password configured in the secret for the administrative user, as described in Section 3.4, "Creating the secret for the administrative user". This user must have the **kie-server,rest-all,admin** roles.
- Clients are created in the RH-SSO authentication system for all components of the Red Hat Decision 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 Decision 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 4.6.1, "Starting configuration of the template for an additional managed KIE Server".

Procedure

1. 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 Decision 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.
2. Complete one of the following procedures:
   a. If you created the client for Red Hat Decision Manager within RH-SSO, set the following parameters in the template:
      - **Business Central RH-SSO Client name** (**DECISION_CENTRAL_SSO_CLIENT**): The RH-SSO client name for Business Central.
      - **KIE Server RH-SSO Client name** (**KIE_SERVER_SSO_CLIENT**): The RH-SSO client name for KIE Server.
      - **KIE Server RH-SSO Client Secret** (**KIE_SERVER_SSO_SECRET**): The secret string that is set in RH-SSO for the client for KIE Server.

   b. To create the clients for Red Hat Decision Manager within RH-SSO, set the following parameters in the template:
      - **KIE Server RH-SSO Client name** (**KIE_SERVER_SSO_CLIENT**): The name of the client to create in RH-SSO for KIE Server.

- **KIE Server RH-SSO Client Secret** (**KIE_SERVER_SSO_SECRET**): The secret string to set in RH-SSO for the client for KIE 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 Decision 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 4.6.9, "Completing deployment of the template for an additional managed KIE Server".

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

4.6.7. Setting parameters for LDAP authentication for an additional managed KIE Server

If you want to use LDAP authentication, complete the following additional configuration when configuring the template to deploy an additional managed KIE Server.

IMPORTANT

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

Prerequisites

- You created user names and passwords for Red Hat Decision Manager in the LDAP system. For a list of the available roles, see Chapter 5, *Red Hat Decision Manager roles and users* .
  You must create a user with the username and password configured in the secret for the administrative user, as described in Section 3.4, "Creating the secret for the administrative user". This user must have the **kie-server,rest-all,admin** roles.
- You started the configuration of the template, as described in Section 4.6.1, "Starting configuration of the template for an additional managed KIE Server".

Procedure

1. 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 Decision 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 4.4, "(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 4.6.9, "Completing deployment of the template for an additional managed KIE Server".

4.6.8. Enabling Prometheus metric collection for an additional managed KIE Server

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

Prerequisites

- You started the configuration of the template, as described in Section 4.6.1, "Starting configuration of the template for an additional managed KIE Server".

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 4.6.9, "Completing deployment of the template for an additional managed KIE Server".

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

4.6.9. Completing deployment of the template for an additional managed KIE Server

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.

CHAPTER 5. RED HAT DECISION MANAGER ROLES AND USERS

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

The Business Central and KIE Server use Java Authentication and Authorization Service (JAAS) login module to authenticate the users. If both Business Central and KIE 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 KIE Server.

However, if Business Central and KIE 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 KIE Server (for example, to view or manage process definitions in Business Central). In case, the user is not authenticated on the KIE 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 Decision Manager user roles.

NOTE

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

- **admin**: Users with the **admin** role are the Business Central administrators. They can manage users and create, clone, and manage 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 Decision Manager.
- **analyst**: Users with the **analyst** role have access to all high-level features. They can model projects. However, these users cannot add contributors to spaces or delete spaces in the **Design → Projects** view. Access to the **Deploy → Execution Servers** view, which is intended for administrators, is not available to users with the **analyst** role. However, the **Deploy** button is available to these users when they access the Library perspective.
- **rest-all**: Users with the **rest-all** role can access Business Central REST capabilities.
- **kie-server**: Users with the **kie-server** role can access KIE Server (KIE Server) REST capabilities.

CHAPTER 6. OPENSHIFT TEMPLATE REFERENCE INFORMATION

Red Hat Decision Manager provides the following OpenShift templates. To access the templates, download and extract the **rhdm-7.8.0-openshift-templates.zip** product deliverable file from the Software Downloads page of the Red Hat customer portal.

- **rhdm78-authoring.yaml** provides a Business Central and a KIE Server connected to the Business Central. You can use this environment to author services and other business assets or to run them in staging or production environments. For details about this template, see Section 6.1, "rhdm78-authoring.yaml template".
- **rhdm78-authoring-ha.yaml** provides a high-availability Business Central and a KIE Server connected to the Business Central. You can use this environment to author services and other business assets or to run them in staging or production environments. For details about this template, see Section 6.2, "rhdm78-authoring-ha.yaml template".
- **rhdm78-kieserver.yaml** provides a KIE Server. You can configure the KIE Server to connect to a Business Central. In this way, you can set up a staging or production environment in which one Business Central manages several distinct KIE Servers. For details about this template, see Section 6.3, "rhdm78-kieserver.yaml template".

6.1. RHDM78-AUTHORING.YAML TEMPLATE

Application template for a non-HA persistent authoring environment, for Red Hat Decision Manager 7.8 - Deprecated

6.1.1. Parameters

Templates allow you to define parameters that 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. See the Openshift documentation for more information.

Variable name	Image Environment Variable	Description	Example value	Required
APPLICATION_NAME	–	The name for the application.	myapp	True
CREDENTIALS_SECRET	–	Secret containing the KIE_ADMIN_USER and KIE_ADMIN_PWD values.	rhpam-credentials	True

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_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_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

Variable name	Image Environment Variable	Description	Example value	Required
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
DECISION_CENTRAL_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route for Decision Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhdmcentr-<project>.<default-domain-suffix>	–	False
DECISION_CENTRAL_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for https service route for Decision Central. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<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

Variable name	Image Environment Variable	Description	Example value	Required
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
DECISION_CENTRAL_HTTPS_SECRET	–	The name of the secret containing the keystore file for Decision Central.	decisioncentral-app-secret	True
DECISION_CENTRAL_HTTPS_KEYSTORE	**HTTPS_KEYSTORE**	The name of the keystore file within the secret.	keystore.jks	False
DECISION_CENTRAL_HTTPS_NAME	**HTTPS_NAME**	The name associated with the server certificate.	jboss	False
DECISION_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.	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
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 Decision Manager images are installed. These ImageStreams are normally installed in the openshift namespace. You need to modify this parameter only 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 "rhdm-kieserver-rhel8".	rhdm-kieserver-rhel8	True
IMAGE_STREAM_TAG	–	A named pointer to an image in an image stream. Default is "7.8.0".	7.8.0	True
MAVEN_MIRROR_URL	**MAVEN_MIRROR_URL**	Maven mirror that Decision 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-rhdmcentr	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-rhdmcentr,!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**	User name for accessing the Maven repository, if required.	–	False
MAVEN_REPO_PASSWORD	**MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	–	False
GIT_HOOKS_DIR	**GIT_HOOKS_DIR**	The directory to use for git hooks, if required.	**/opt/kie/data/git/hooks**	False
DECISION_CENTRAL_VOLUME_CAPACITY	–	Size of the persistent storage for Decision Central's runtime data.	1Gi	True

Variable name	Image Environment Variable	Description	Example value	Required
DECISION_CENTRAL_MEMORY_LIMIT	–	Decision 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
DECISION_CENTRAL_SSO_CLIENT	**SSO_CLIENT**	Decision Central RH-SSO Client name	–	False
DECISION_CENTRAL_SSO_SECRET	**SSO_SECRET**	Decision 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 user name 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

Variable name	Image Environment Variable	Description	Example value	Required
SSO_PRINCIPAL_ATTRIBUTE	**SSO_PRINCIPAL_ATTRIBUTE**	RH-SSO Principal Attribute to use as user name.	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=example,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
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

Variable name	Image Environment Variable	Description	Example value	Required
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
AUTH_LDAP_PARSE_USERNAME	**AUTH_LDAP_PARSE_USERNAME**	A flag indicating if the DN is to be parsed for the user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. This option is used together with usernameBeginString and usernameEndString.	true	False

Variable name	Image Environment Variable	Description	Example value	Required
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 user name. 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 user name. 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 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

6.1.2. 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.

6.1.2.1. Services

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

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

6.1.2.2. 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. See the Openshift documentation for more information.

Service	Security	Hostname
insecure-${APPLICATION_NAME}-rhdmcentr-http	none	**${DECISION_CENTRAL_HOSTNAME_HTTP}**
${APPLICATION_NAME}-rhdmcentr-https	TLS passthrough	**${DECISION_CENTRAL_HOSTNAME_HTTPS}**
insecure-${APPLICATION_NAME}-kieserver-http	none	**${KIE_SERVER_HOSTNAME_HTTP}**
${APPLICATION_NAME}-kieserver-https	TLS passthrough	**${KIE_SERVER_HOSTNAME_HTTPS}**

6.1.2.3. 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. See the Openshift documentation for more information.

6.1.2.3.1. Triggers

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

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

6.1.2.3.2. 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. See the container-engine documentation for more information.

Deployment	Replicas
${APPLICATION_NAME}-rhdmcentr	1
${APPLICATION_NAME}-kieserver	1

6.1.2.3.3. Pod Template

6.1.2.3.3.1. Service Accounts

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

Deployment	Service Account
${APPLICATION_NAME}-rhdmcentr	**${APPLICATION_NAME}-rhdmsvc**
${APPLICATION_NAME}-kieserver	**${APPLICATION_NAME}-rhdmsvc**

6.1.2.3.3.2. Image

Deployment	Image
${APPLICATION_NAME}-rhdmcentr	rhdm-decisioncentral-rhel8
${APPLICATION_NAME}-kieserver	**${KIE_SERVER_IMAGE_STREAM_NAME}**

6.1.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-rhdmcentr

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

${APPLICATION_NAME}-kieserver

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

6.1.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-rhdmcentr

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

${APPLICATION_NAME}-kieserver

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

6.1.2.3.3.5. Exposed Ports

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

6.1.2.3.3.6. Image Environment Variables

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

Deployment	Variable name	Description	Example value
	KIE_ADMIN_USER	Admin user name	Set according to the credentials secret
	KIE_ADMIN_PWD	Admin user password	Set according to the credentials secret
	KIE_MBEANS	KIE server mbeans enabled/disabled (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_CONTROLLER_OPENSHIFT_ENABLED	–	false
	KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED	If set to true, turns on KIE server global discovery feature (Sets the org.kie.server.controller.openshift.global.discovery.enabled system property)	**${KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED}**
	KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	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}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication. (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	WORKBENCH_ROUTE_NAME	–	**${APPLICATION_NAME}-rhdmcentr**
	MAVEN_MIRROR_URL	Maven mirror that Decision 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.	**${MAVEN_MIRROR_URL}**
	MAVEN_REPO_ID	The id to use for the maven repository. If set, it can be excluded from the optionally configured mirror by adding it to MAVEN_MIRROR_OF. For example: external:*,!repo-rhdmcentr,!repo-custom. If MAVEN_MIRROR_URL is set but MAVEN_MIRROR_ID is not set, an id will be generated randomly, but won't be usable in MAVEN_MIRROR_OF.	**${MAVEN_REPO_ID}**
	MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	MAVEN_REPO_USERNAME	User name for accessing the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**

Deployment	Variable name	Description	Example value
	GIT_HOOKS_DIR	The directory to use for git hooks, if required.	**${GIT_HOOKS_DIR}**
	HTTPS_KEYSTORE_DIR	–	**/etc/decisioncentral-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret.	**${DECISION_CENTRAL_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate.	**${DECISION_CENTRAL_HTTPS_NAME}**
	HTTPS_PASSWORD	The password for the keystore and certificate.	**${DECISION_CENTRAL_HTTPS_PASSWORD}**
	SSO_URL	RH-SSO URL.	**${SSO_URL}**
	SSO_OPENIDCONNECT_DEPLOYMENTS	–	ROOT.war
	SSO_REALM	RH-SSO Realm name.	**${SSO_REALM}**
	SSO_SECRET	Decision Central RH-SSO Client Secret.	**${DECISION_CENTRAL_SSO_SECRET}**
	SSO_CLIENT	Decision Central RH-SSO Client name	**${DECISION_CENTRAL_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm admin user name 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}**

Deployment	Variable name	Description	Example value
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as user name.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route for Decision Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhdmcentr-<project>.<default-domain-suffix>	**${DECISION_CENTRAL_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for https service route for Decision Central. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<project>.<default-domain-suffix>	**${DECISION_CENTRAL_HOSTNAME_HTTPS}**
	AUTH_LDAP_URL	LDAP Endpoint to connect for authentication.	**${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 user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. 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 user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${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}-rhdmcentr**
	KIE_ADMIN_USER	Admin user name	Set according to the credentials secret
	KIE_ADMIN_PWD	Admin user password	Set according to the credentials secret
	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}**

Deployment	Variable name	Description	Example value
	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)	**${KIE_SERVER_BYPASS_AUTH_USER}**
	KIE_SERVER_CONTROLLER_SERVICE	–	**${APPLICATION_NAME}-rhdmcentr**
	KIE_SERVER_CONTROLLER_PROTOCOL	–	ws
	KIE_SERVER_ID	–	–
	KIE_SERVER_ROUTE_NAME	–	insecure-${APPLICATION_NAME}-kieserver
	KIE_SERVER_STARTUP_STRATEGY	–	ControllerBasedStartupStrategy
	MAVEN_MIRROR_URL	Maven mirror that Decision Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	**${MAVEN_MIRROR_URL}**

Deployment	Variable name	Description	Example value
	MAVEN_MIRROR_OF	Maven mirror configuration for KIE server.	**${MAVEN_MIRROR_OF}**
	MAVEN_REPOS	–	RHDMCENTR,EXTERNAL
	RHDMCENTR_MAVEN_REPO_ID	–	repo-rhdmcentr
	RHDMCENTR_MAVEN_REPO_SERVICE	–	**${APPLICATION_NAME}-rhdmcentr**
	RHDMCENTR_MAVEN_REPO_PATH	–	**/maven2/**
	RHDMCENTR_MAVEN_REPO_USERNAME	–	Set according to the credentials secret
	RHDMCENTR_MAVEN_REPO_PASSWORD	–	Set according to the credentials secret
	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-rhdmcentr,!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}**

Deployment	Variable name	Description	Example value
	EXTERNAL_MAVEN_REPO_USERNAME	User name for accessing 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.	**${KIE_SERVER_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate.	**${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.	**${KIE_SERVER_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm admin user name used to create the Client if it doesn't exist.	**${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.	**${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}**
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as user name.	**${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}**
	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 user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. 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 user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${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}**

6.1.2.3.3.7. Volumes

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

6.1.2.4. External Dependencies

6.1.2.4.1. 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. See the Openshift documentation for more information.

Name	Access Mode
${APPLICATION_NAME}-rhdmcentr-claim	ReadWriteOnce

6.1.2.4.2. Secrets

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

decisioncentral-app-secret kieserver-app-secret

6.2. RHDM78-AUTHORING-HA.YAML TEMPLATE

Application template for a HA persistent authoring environment, for Red Hat Decision Manager 7.8 - Deprecated

6.2.1. Parameters

Templates allow you to define parameters that 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. See the Openshift documentation for more information.

Variable name	Image Environment Variable	Description	Example value	Required
APPLICATION_NAME	–	The name for the application.	myapp	True
CREDENTIALS_SECRET	–	Secret containing the KIE_ADMIN_USER and KIE_ADMIN_PWD values.	rhpam-credentials	True
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_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

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
DECISION_CENTRAL_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route for Decision Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhdmcentr-<project>.<default-domain-suffix>	–	False
DECISION_CENTRAL_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for https service route for Decision Central. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<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
DECISION_CENTRAL_HTTPS_SECRET	–	The name of the secret containing the keystore file for Decision Central.	decisioncentral-app-secret	True
DECISION_CENTRAL_HTTPS_KEYSTORE	**HTTPS_KEYSTORE**	The name of the keystore file within the secret for Decision Central.	keystore.jks	False
DECISION_CENTRAL_HTTPS_NAME	**HTTPS_NAME**	The name associated with the server certificate for Decision Central.	jboss	False
DECISION_CENTRAL_HTTPS_PASSWORD	**HTTPS_PASSWORD**	The password for the keystore and certificate for Decision 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 user name 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:1.5	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:7.6	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

Variable name	Image Environment Variable	Description	Example value	Required
AMQ_VOLUME_CAPACITY	–	Size of persistent storage for AMQ broker volume.	1Gi	True
AMQ_REPLICAS	–	Number of broker replicas for a cluster	2	True
DECISION_CENTRAL_CONTAINER_REPLICAS	–	Decision Central Container Replicas, defines how many Decision Central containers will be started.	2	True
KIE_SERVER_CONTAINER_REPLICAS	–	KIE Server Container Replicas, defines how many KIE Server containers will be started.	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**	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

Variable name	Image Environment Variable	Description	Example value	Required
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
IMAGE_STREAM_NAMESPACE	–	Namespace in which the ImageStreams for Red Hat Decision Manager images are installed. These ImageStreams are normally installed in the openshift namespace. You need to modify this parameter only if you installed the ImageStreams in a different namespace/project.	openshift	True
DECISION_CENTRAL_IMAGE_STREAM_NAME	–	The name of the image stream to use for Decision Central. Default is "rhdm-decisioncentral-rhel8".	rhdm-decisioncentral-rhel8	True
KIE_SERVER_IMAGE_STREAM_NAME	–	The name of the image stream to use for KIE server. Default is "rhdm-kieserver-rhel8".	rhdm-kieserver-rhel8	True
IMAGE_STREAM_TAG	–	A named pointer to an image in an image stream. Default is "7.8.0".	7.8.0	True

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_MIRROR_URL	**MAVEN_MIRROR_URL**	Maven mirror that Decision 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-rhdmcentr	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-rhdmcentr,!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

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_REPO_USERNAME	**MAVEN_REPO_USERNAME**	User name for accessing the Maven repository, if required.	–	False
MAVEN_REPO_PASSWORD	**MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	–	False
GIT_HOOKS_DIR	**GIT_HOOKS_DIR**	The directory to use for git hooks, if required.	**/opt/kie/data/git/hooks**	False
DECISION_CENTRAL_VOLUME_CAPACITY	–	Size of the persistent storage for Decision Central's runtime data.	1Gi	True
DECISION_CENTRAL_MEMORY_LIMIT	–	Decision Central Container memory limit.	8Gi	True
DECISION_CENTRAL_JAVA_MAX_MEM_RATIO	**JAVA_MAX_MEM_RATIO**	Decision Central Container JVM max memory ratio. **-Xmx** is set to a ratio of the memory available on the container. The default is 80, which means the upper boundary is 80% of the available memory. To skip adding the **-Xmx** option, set this value to 0.	80	True
DECISION_CENTRAL_CPU_LIMIT	–	Decision Central Container CPU limit.	2000m	True
KIE_SERVER_MEMORY_LIMIT	–	KIE server Container memory limit.	1Gi	True

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_CPU_LIMIT	–	KIE server Container CPU limit.	1000m	True
SSO_URL	**SSO_URL**	RH-SSO URL.	https://rh-sso.example.com/auth	False
SSO_REALM	**SSO_REALM**	RH-SSO Realm name.	–	False
DECISION_CENTRAL_SSO_CLIENT	**SSO_CLIENT**	Decision Central RH-SSO Client name.	–	False
DECISION_CENTRAL_SSO_SECRET	**SSO_SECRET**	Decision 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 user name 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 user name.	preferred_username	False

Variable name	Image Environment Variable	Description	Example value	Required
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=example,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
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 user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. 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 user name. 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 user name. 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 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

6.2.2. 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.

6.2.2.1. Services

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

Service	Port	Name	Description
${APPLICATION_NAME}-rhdmcentr	8080	http	All the Decision Central web server's ports.
	8443	https	
${APPLICATION_NAME}-rhdmcentr-ping	8888	ping	The JGroups ping port for rhdmcentr clustering.
${APPLICATION_NAME}-datagrid-ping	8888	ping	Provides a ping service for clustered applications.
${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.

6.2.2.2. 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. See the Openshift documentation for more information.

Service	Security	Hostname
insecure-${APPLICATION_NAME}-rhdmcentr-http	none	**${DECISION_CENTRAL_HOSTNAME_HTTP}**
${APPLICATION_NAME}-rhdmcentr-https	TLS passthrough	**${DECISION_CENTRAL_HOSTNAME_HTTPS}**
insecure-${APPLICATION_NAME}-kieserver-http	none	**${KIE_SERVER_HOSTNAME_HTTP}**
${APPLICATION_NAME}-kieserver-https	TLS passthrough	**${KIE_SERVER_HOSTNAME_HTTPS}**

6.2.2.3. 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. See the Openshift documentation for more information.

6.2.2.3.1. Triggers

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

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

6.2.2.3.2. 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. See the container-engine documentation for more information.

Deployment	Replicas
${APPLICATION_NAME}-rhdmcentr	2
${APPLICATION_NAME}-kieserver	2

6.2.2.3.3. Pod Template

6.2.2.3.3.1. Service Accounts

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

Deployment	Service Account
${APPLICATION_NAME}-rhdmcentr	**${APPLICATION_NAME}-rhdmsvc**
${APPLICATION_NAME}-kieserver	**${APPLICATION_NAME}-rhdmsvc**

6.2.2.3.3.2. Image

Deployment	Image
${APPLICATION_NAME}-rhdmcentr	**${DECISION_CENTRAL_IMAGE_STREAM_NAME}**
${APPLICATION_NAME}-kieserver	**${KIE_SERVER_IMAGE_STREAM_NAME}**

6.2.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-rhdmcentr

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

${APPLICATION_NAME}-kieserver

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

6.2.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-rhdmcentr

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

${APPLICATION_NAME}-kieserver

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

6.2.2.3.3.5. Exposed Ports

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

6.2.2.3.3.6. Image Environment Variables

Deployment	Variable name	Description	Example value
${APPLICATION_NAME}-rhdmcentr	**APPLICATION_USERS_PROPERTIES**	–	**/opt/kie/data/configuration/application-users.properties**
	APPLICATION_ROLES_PROPERTIES	–	**/opt/kie/data/configuration/application-roles.properties**
	KIE_ADMIN_USER	Admin user name	Set according to the credentials secret
	KIE_ADMIN_PWD	Admin user password	Set according to the credentials secret
	KIE_MBEANS	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_CONTROLLER_OPENSHIFT_ENABLED	–	true

Deployment	Variable name	Description	Example value
	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)	**${KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED}**
	KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	If OpenShift integration of Business Central is turned on, setting this parameter to true enables connection to KIE Server via an OpenShift internal Service endpoint. (Sets the org.kie.server.controller.openshift.prefer.kieserver.service system property)	**${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)	**${KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL}**
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication. (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	WORKBENCH_ROUTE_NAME	–	**${APPLICATION_NAME}-rhdmcentr**
	MAVEN_MIRROR_URL	Maven mirror that Decision Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	**${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-rhdmcentr,!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	User name for accessing the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	GIT_HOOKS_DIR	The directory to use for git hooks, if required.	**${GIT_HOOKS_DIR}**
	HTTPS_KEYSTORE_DIR	–	**/etc/decisioncentral-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret for Decision Central.	**${DECISION_CENTRAL_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate for Decision Central.	**${DECISION_CENTRAL_HTTPS_NAME}**

Deployment	Variable name	Description	Example value
	HTTPS_PASSWORD	The password for the keystore and certificate for Decision Central.	**${DECISION_CENTRAL_HTTPS_PASSWORD}**
	JGROUPS_PING_PROTOCOL	–	openshift.DNS_PING
	OPENSHIFT_DNS_PING_SERVICE_NAME	–	**${APPLICATION_NAME}-rhdmcentr-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 user name 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}**
	JAVA_MAX_MEM_RATIO	Decision Central Container JVM max memory ratio. **-Xmx** is set to a ratio of the memory available on the container. The default is 80, which means the upper boundary is 80% of the available memory. To skip adding the **-Xmx** option, set this value to 0.	**${DECISION_CENTRAL_JAVA_MAX_MEM_RATIO}**
	SSO_URL	RH-SSO URL.	**${SSO_URL}**

Deployment	Variable name	Description	Example value
	SSO_OPENIDCONNECT_DEPLOYMENTS	–	ROOT.war
	SSO_REALM	RH-SSO Realm name.	**${SSO_REALM}**
	SSO_SECRET	Decision Central RH-SSO Client Secret.	**${DECISION_CENTRAL_SSO_SECRET}**
	SSO_CLIENT	Decision Central RH-SSO Client name.	**${DECISION_CENTRAL_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm admin user name used to create the Client if it doesn't exist.	**${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 user name.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route for Decision Central. Leave blank for default hostname, e.g.: insecure-<application-name>-rhdmcentr-<project>.<default-domain-suffix>	**${DECISION_CENTRAL_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for https service route for Decision Central. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<project>.<default-domain-suffix>	**${DECISION_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.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. This option is used together with usernameBeginString and usernameEndString.	**${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 user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_END_STRING}**

Deployment	Variable name	Description	Example value
	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}**
	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.	**${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}**
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format is original_role=role1,role2,role3	**${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}-rhdmcentr**
	KIE_ADMIN_USER	Admin user name	Set according to the credentials secret
	KIE_ADMIN_PWD	Admin user password	Set according to the credentials secret

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).	**${KIE_SERVER_MODE}**
	KIE_MBEANS	KIE server mbeans enabled/disabled. (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${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}**
	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)	**${KIE_SERVER_BYPASS_AUTH_USER}**
	KIE_SERVER_CONTROLLER_SERVICE	–	**${APPLICATION_NAME}-rhdmcentr**
	KIE_SERVER_CONTROLLER_PROTOCOL	–	ws

Deployment	Variable name	Description	Example value
	KIE_SERVER_ID	–	–
	KIE_SERVER_ROUTE_NAME	–	insecure-${APPLICATION_NAME}-kieserver
	KIE_SERVER_STARTUP_STRATEGY	–	OpenShiftStartupStrategy
	MAVEN_MIRROR_URL	Maven mirror that Decision Central and KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for building and deploying your services.	**${MAVEN_MIRROR_URL}**
	MAVEN_MIRROR_OF	Maven mirror configuration for KIE server.	**${MAVEN_MIRROR_OF}**
	MAVEN_REPOS	–	RHDMCENTR,EXTERNAL
	RHDMCENTR_MAVEN_REPO_ID	–	repo-rhdmcentr
	RHDMCENTR_MAVEN_REPO_SERVICE	–	**${APPLICATION_NAME}-rhdmcentr**
	RHDMCENTR_MAVEN_REPO_PATH	–	**/maven2/**
	RHDMCENTR_MAVEN_REPO_USERNAME	–	Set according to the credentials secret
	RHDMCENTR_MAVEN_REPO_PASSWORD	–	Set according to the credentials secret

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-rhdmcentr,!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	User name for accessing 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.	**${KIE_SERVER_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm admin user name used to create the Client if it doesn't exist.	**${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 user name.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route for KIE Server. Leave blank for default hostname, e.g.: insecure-<application-name>-kieserver-<project>.<default-domain-suffix>	**${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.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. This option is used together with usernameBeginString and usernameEndString.	**${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 user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${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.	**${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}**
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This parameter defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format is original_role=role1,role2,role3	**${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}**

6.2.2.3.3.7. Volumes

Deployment	Name	mountPath	Purpose	readOnly
${APPLICATION_NAME}-rhdmcentr	decisioncentral-keystore-volume	**/etc/decisioncentral-secret-volume**	ssl certs	True

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

6.2.2.4. External Dependencies

6.2.2.4.1. 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. See the Openshift documentation for more information.

Name	Access Mode
${APPLICATION_NAME}-rhdmcentr-claim	ReadWriteMany

6.2.2.4.2. Secrets

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

decisioncentral-app-secret kieserver-app-secret

6.2.2.4.3. 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 6.1. 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
```

6.3. RHDM78-KIESERVER.YAML TEMPLATE

Application template for a managed KIE Server, for Red Hat Decision Manager 7.8 - Deprecated

6.3.1. Parameters

Templates allow you to define parameters that 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. See the Openshift documentation for more information.

Variable name	Image Environment Variable	Description	Example value	Required
APPLICATION_NAME	–	The name for the application.	myapp	True
MAVEN_MIRROR_URL	**MAVEN_MIRROR_URL**	Maven mirror that KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for deploying your services.	–	False
MAVEN_MIRROR_OF	**MAVEN_MIRROR_OF**	Maven mirror configuration for KIE server.	external:*	False
MAVEN_REPO_ID	**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-rhdmcentr,!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	**EXTERNAL_MAVEN_REPO_URL**	Fully qualified URL to a Maven repository or service.	http://nexus.nexus-project.svc.cluster.local:8081/nexus/content/groups/public/	True

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_REPO_USERNAME	**EXTERNAL_MAVEN_REPO_USERNAME**	User name for accessing the Maven repository, if required.	–	False
MAVEN_REPO_PASSWORD	**EXTERNAL_MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	–	False
DECISION_CENTRAL_SERVICE	**WORKBENCH_SERVICE_NAME**	The Service name for the optional Decision Central, where it can be reached, to allow service lookups (for example, maven repo usage), if required.	myapp-rhdmcentr	False
CREDENTIALS_SECRET	–	Secret containing the KIE_ADMIN_USER and KIE_ADMIN_PWD values.	rhpam-credentials	True
IMAGE_STREAM_NAMESPACE	–	Namespace in which the ImageStreams for Red Hat Decision Manager images are installed. These ImageStreams are normally installed in the openshift namespace. You need to modify this parameter only if you installed the ImageStreams in a different namespace/project.	openshift	True

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_IMAGE_STREAM_NAME	–	The name of the image stream to use for KIE server. Default is "rhdm-kieserver-rhel8".	rhdm-kieserver-rhel8	True
IMAGE_STREAM_TAG	–	A named pointer to an image in an image stream. Default is "7.8.0".	7.8.0	True
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).	**PRODUCTION**	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
KIE_SERVER_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route. 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. Leave blank for default hostname, e.g.: <application-name>-kieserver-<project>.<default-domain-suffix>	–	False
KIE_SERVER_HTTPS_SECRET	–	The name of the secret containing the keystore file.	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
KIE_SERVER_HTTPS_PASSWORD	**HTTPS_PASSWORD**	The password for the keystore and certificate.	mykeystorepass	False

Variable name	Image Environment Variable	Description	Example value	Required
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_MEMORY_LIMIT	–	KIE server Container memory limit.	1Gi	False
KIE_SERVER_CONTAINER_DEPLOYMENT	**KIE_SERVER_CONTAINER_DEPLOYMENT**	KIE Server Container deployment configuration with optional alias. Format: containerId=groupId:artifactId:version |c2(alias2)=g2:a2:v2	rhdm-kieserver-library=org.openshift.quickstarts:rhdm-kieserver-library:1.6.0-SNAPSHOT	False
KIE_SERVER_MGMT_DISABLED	**KIE_SERVER_MGMT_DISABLED**	Disable management api and don't allow KIE containers to be deployed/undeployed or started/stopped. Sets the property org.kie.server.mgmt.api.disabled to true and org.kie.server.startup.strategy to LocalContainersStartupStrategy.	true	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
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 user name 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 user name.	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=example,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

Variable name	Image Environment Variable	Description	Example value	Required
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
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 user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. 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 user name. 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 user name. 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 property 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

6.3.2. 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.

6.3.2.1. Services

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

Service	Port	Name	Description
${APPLICATION_NAME}-kieserver	8080	http	All the KIE server web server's ports.
	8443	https	
${APPLICATION_NAME}-kieserver-ping	8888	ping	The JGroups ping port for clustering.

6.3.2.2. 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. See the Openshift documentation for more information.

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

6.3.2.3. 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. See the Openshift documentation for more information.

6.3.2.3.1. Triggers

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

Deployment	Triggers
${APPLICATION_NAME}-kieserver	ImageChange

6.3.2.3.2. 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. See the container-engine documentation for more information.

Deployment	Replicas
${APPLICATION_NAME}-kieserver	1

6.3.2.3.3. Pod Template

6.3.2.3.3.1. Service Accounts

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

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

6.3.2.3.3.2. Image

Deployment	Image
${APPLICATION_NAME}-kieserver	**${KIE_SERVER_IMAGE_STREAM_NAME}**

6.3.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-kieserver

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

6.3.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-kieserver

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

6.3.2.3.3.5. Exposed Ports

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

6.3.2.3.3.6. Image Environment Variables

Deployment	Variable name	Description	Example value
${APPLICATION_NAME}-kieserver	**WORKBENCH_SERVICE_NAME**	The Service name for the optional Decision Central, where it can be reached, to allow service lookups (for example, maven repo usage), if required.	**${DECISION_CENTRAL_SERVICE}**

Deployment	Variable name	Description	Example value
	KIE_ADMIN_USER	Admin user name	Set according to the credentials secret
	KIE_ADMIN_PWD	Admin user password	Set according to the credentials secret
	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}**
	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	–	–

Deployment	Variable name	Description	Example value
	KIE_SERVER_ROUTE_NAME	–	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_CONTAINER_DEPLOYMENT	KIE Server Container deployment configuration with optional alias. Format: containerId=groupId:artifactId:version|c2(alias2)=g2:a2:v2	**${KIE_SERVER_CONTAINER_DEPLOYMENT}**
	MAVEN_MIRROR_URL	Maven mirror that KIE server must use. If you configure a mirror, this mirror must contain all artifacts that are required for deploying your services.	**${MAVEN_MIRROR_URL}**
	MAVEN_MIRROR_OF	Maven mirror configuration for KIE server.	**${MAVEN_MIRROR_OF}**
	MAVEN_REPOS	–	RHDMCENTR,EXTERNAL
	RHDMCENTR_MAVEN_REPO_ID	–	repo-rhdmcentr
	RHDMCENTR_MAVEN_REPO_SERVICE	The Service name for the optional Decision Central, where it can be reached, to allow service lookups (for example, maven repo usage), if required.	**${DECISION_CENTRAL_SERVICE}**
	RHDMCENTR_MAVEN_REPO_PATH	–	**/maven2/**
	RHDMCENTR_MAVEN_REPO_USERNAME	–	Set according to the credentials secret
	RHDMCENTR_MAVEN_REPO_PASSWORD	–	Set according to the credentials secret

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-rhdmcentr,!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	User name for accessing the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	EXTERNAL_MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	KIE_SERVER_MGMT_DISABLED	Disable management api and don't allow KIE containers to be deployed/undeployed or started/stopped. Sets the property org.kie.server.mgmt.api.disabled to true and org.kie.server.startup.strategy to LocalContainersStartupStrategy.	**${KIE_SERVER_MGMT_DISABLED}**
	KIE_SERVER_STARTUP_STRATEGY	–	OpenShiftStartupStrategy
	HTTPS_KEYSTORE_DIR	–	**/etc/kieserver-secret-volume**

Deployment	Variable name	Description	Example value
	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.	**${KIE_SERVER_HTTPS_PASSWORD}**
	JGROUPS_PING_PROTOCOL	–	openshift.DNS_PING
	OPENSHIFT_DNS_PING_SERVICE_NAME	–	**${APPLICATION_NAME}-kieserver-ping**
	OPENSHIFT_DNS_PING_SERVICE_PORT	–	8888
	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 user name 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}**

Deployment	Variable name	Description	Example value
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as user name.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route. Leave blank for default hostname, e.g.: insecure-<application-name>-kieserver-<project>.<default-domain-suffix>	**${KIE_SERVER_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for https service route. 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 user name. If set to true, the DN is parsed for the user name. If set to false the DN is not parsed for the user name. 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 user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the user name. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${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 property 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}**

6.3.2.3.3.7. Volumes

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

6.3.2.4. External Dependencies

6.3.2.4.1. Secrets

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

kieserver-app-secret

6.4. OPENSHIFT USAGE QUICK REFERENCE

To deploy, monitor, manage, and undeploy Red Hat Decision 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 Thursday, September 08, 2020.