Red Hat Decision Manager 7.2

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

Last Updated: 2019-02-25

Red Hat Decision Manager 7.2 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 7.2 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

- At least four gigabytes of memory must be available in the OpenShift cluster/namespace.
- The OpenShift project for the deployment must be created.
- You must be 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 must be enabled. Alternatively, if dynamic PV provisioning is not enabled, a sufficient persistent volume must be available. By default, Decision Central requires one 1Gi PV. You can change the PV size for Decision Central persistent storage in the template parameters.
- If you intend to use the Authoring High Availability template, which scales the Decision Central pod:
  - The image streams for Red Hat AMQ version 7.1 or later must be available in your OpenShift environment.
  - Your OpenShift environment must support persistent volumes with ReadWriteMany mode. For information about access mode support in OpenShift Online volume plug-ins, see Access Modes.

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, providing as few or as many containers as necessary for a particular component. You can use standard OpenShift methods to manage the pods and balance the load.

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

- Decision Server, also known as *Execution Server* or *KIE 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.
  You can freely scale up a Decision Server pod, providing as many copies as necessary, running on the same host or different hosts. As you scale a pod up or down, all 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 Decision 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 Decision Server pods as necessary.

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

  Decision 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, Decision Central can place the compiled services (KJAR files) into the built-in Maven repository or (if configured) into an external Maven repository.

IMPORTANT

In the current version, high-availability Decision Central functionality is a technology preview.

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 Decision Central and also for running services on Decision Servers. It consists of pods that provide Decision Central for the authoring work and one or more Decision Servers for execution of the services. Each Decision Server is a pod that you can replicate by scaling it up or down as necessary. You can deploy and undeploy services on each Decision Server using Decision 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 Decision 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 Decision 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 Decision Central and a Decision 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. PREPARING TO DEPLOY RED HAT DECISION MANAGER IN YOUR OPENSHIFT ENVIRONMENT

Before deploying Red Hat Decision Manager in your OpenShift environment, you need to complete several preparatory 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

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

To deploy Red Hat Decision Manager components of 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 the information about their location (known as *image streams*). 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 the same project.

Procedure

1. Determine whether Red Hat OpenShift Container Platform was 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 was configured with the user name and password for Red Hat registry access, run the following commands:

   ```
   $ oc get imagestreamtag -n openshift | grep rhdm72-decisioncentral-
   openshift
   $ oc get imagestreamtag -n openshift | grep rhdm72-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 was 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 on to 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. Run 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
```

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

f. Download the **`rhdm-7.2.0-openshift-templates.zip`** product deliverable file from the Software Downloads page and extract the **`rhdm72-image-streams.yaml`** file.

g. Complete one of the following actions:

- Run the following command:

```
$ oc create -f rhdm72-image-streams.yaml
```

- Using the OpenShift Web UI, select **Add to Project** → **Import YAML / JSON** and then choose the file or paste its contents.

NOTE

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

2.2. CREATING THE SECRETS FOR DECISION SERVER

OpenShift uses objects called **`Secrets`** to hold sensitive information, such as passwords or keystores. See the Secrets chapter in the OpenShift documentation for more information.

You must create an SSL certificate for Decision 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 Decision Server. In a production environment, generate a valid signed certificate that matches the expected URL of the Decision Server. Save the keystore in a file named **`keystore.jks`**. Record the name of the certificate and the password of the keystore file.
   See Generate a SSL Encryption Key and Certificate for more information on how to create a keystore with self-signed or purchased SSL certificates.

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

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

2.3. CREATING THE SECRETS FOR DECISION CENTRAL

If you are planning to deploy Decision Central in your OpenShift environment, you must create an SSL certificate for Decision Central and provide it to your OpenShift environment as a secret. Do not use the same certificate and keystore for Decision Central and for Decision Server.

Procedure

1. Generate an SSL keystore with a private and public key for SSL encryption for Decision Central. In a production environment, generate a valid signed certificate that matches the expected URL of the Decision Central. Save the keystore in a file named **`keystore.jks`**. Record the name of the certificate and the password of the keystore file.
   See Generate a SSL Encryption Key and Certificate for more information on how to create a keystore with self-signed or purchased SSL certificates.

2. 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
```

2.4. CHANGING GLUSTERFS CONFIGURATION

Check whether your OpenShift environment uses GlusterFS to provide permanent storage volumes. If it uses GlusterFS, to ensure optimal performance, tune your GlusterFS storage by changing the storage class configuration.

Procedure

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

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

   Where **`class-name`** is the name of the default storage class. For 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. On the line with the **volumeoptions** key, add the following two 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
```

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

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

   Where **class-name** is the name of the default storage class. For example:

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

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

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

CHAPTER 3. AUTHORING OR MANAGED SERVER ENVIRONMENT

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

You can use Decision Central both to develop services and to deploy them to one or several Decision Servers. For example, you can deploy test versions of services to one Decision Server and production versions to another Decision Server.

To avoid accidentally deploying wrong versions to a production Decision Server, you can create separate environments to author services (*authoring environment*) and to manage deployment of production services (*managed server environment*). You can use a shared external Maven repository between these environments, so that services developed in the authoring environment are available in the managed server environment. However, the procedures to deploy these environments are the same.

Depending on your needs, you can deploy either a single or high-availability Decision Central. A single Decision Central pod is not replicated; only a single copy of Decision Central is used. In an HA Decision Central deployment, you can scale Decision Central.

An HA Decision Central provides maximum reliability and responsiveness for authoring services, but has higher memory and storage requirements and also requires support for persistent volumes with ReadWriteMany mode.

IMPORTANT

In the current version, the high-availability functionality is a technology preview.

You can scale Decision Server pods as necessary in any version of the authoring or managed server environment.

To deploy an authoring or managed server environment, first deploy the single or high-availability Decision Central and a single Decision Server using the authoring template.

To add additional Decision Servers, you can deploy the Decision Server template in the same project.

3.1. DEPLOYING SINGLE DECISION CENTRAL AND ONE DECISION SERVER IN AN AUTHORING OR MANAGED SERVER ENVIRONMENT

To deploy single Decision Central and one Decision Server in an authoring or managed server environment, use the **`rhdm72-authoring.yaml`** template file. You can extract this file from the **`rhdm-7.2.0-openshift-templates.zip`** product deliverable file. You can download the file from the Software Downloads page.

Procedure

1. Use one of the following methods to deploy the template:
    - In the OpenShift Web UI, select **Add to Project** → **Import YAML / JSON** and then select or paste the **`rhdm72-authoring.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>/rhdm72-authoring.yaml -p
DECISION_CENTRAL_HTTPS_SECRET=decisioncentral-app-secret -p
KIE_SERVER_HTTPS_SECRET=kieserver-app-secret
```

In this command line:

- 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. You can view the template file to see descriptions for all parameters.

2. Set the following parameters as necessary:

- **Decision Central Server Keystore Secret Name** (**`DECISION_CENTRAL_HTTPS_SECRET`**): The name of the secret for Decision Central, as created in Section 2.3, “Creating the secrets for Decision Central”.
- **KIE Server Keystore Secret Name** (**`KIE_SERVER_HTTPS_SECRET`**): The name of the secret for Decision Server, as created in Section 2.2, “Creating the secrets for Decision Server”.
- **Application Name** (**`APPLICATION_NAME`**): The name of the OpenShift application. It is used in the default URLs for Decision Central and Decision 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) on the Decision Central that the Decision Server is to join.
- **Decision Central Server Certificate Name** (**`DECISION_CENTRAL_HTTPS_NAME`**): The name of the certificate in the keystore that you created in Section 2.3, “Creating the secrets for Decision Central”.
- **Decision Central Server Keystore Password** (**`DECISION_CENTRAL_HTTPS_PASSWORD`**): The password for the keystore that you created in Section 2.3, “Creating the secrets for Decision Central”.
- **KIE Server Certificate Name** (**`KIE_SERVER_HTTPS_NAME`**): The name of the certificate in the keystore that you created in Section 2.2, “Creating the secrets for Decision Server”.
- **KIE Server Keystore Password** (**`KIE_SERVER_HTTPS_PASSWORD`**): The password for the keystore that you created in Section 2.2, “Creating the secrets for Decision Server”.
- **ImageStream Namespace** (**`IMAGE_STREAM_NAMESPACE`**): The namespace where the image streams are available. If the image streams were already available in your OpenShift environment (see Section 2.1, “Ensuring the availability of image streams and the image registry”), the namespace is **`openshift`**. If you have installed the image streams file, the namespace is the name of the OpenShift project.
  You can also set the following user names and passwords:
- **KIE Admin User** (**`KIE_ADMIN_USER`**) and **KIE Admin Password** (**`KIE_ADMIN_PWD`**): The user name and password for the administrative user in Decision Central.
- **KIE Server User** (**`KIE_SERVER_USER`**) and **KIE Server Password** (**`KIE_SERVER_PWD`**): The user name and password that a client application must use to connect to the Decision Server.

3. If you want to deploy additional Decision Servers and connect them to this Decision Central, set the following parameters:

   - **KIE Server Controller User** (**`KIE_SERVER_CONTROLLER_USER`**) and **KIE Server Controller Password** (**`KIE_SERVER_CONTROLLER_PWD`**): The user name and password that a Decision Server must use to connect to the Decision Central.

4. If you want to place the built KJAR files into an external Maven repository, set the following parameters:

   - **Maven repository URL** (**`MAVEN_REPO_URL`**): The URL for the Maven repository.
   - **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.
   - **Maven repository ID** (**`MAVEN_REPO_ID`**): The Maven ID, which must match the **`id`** setting for the Maven repository.
     Alternatively, if you want to use the Maven repository that is built into Decision Central and to connect additional Decision Servers to the Decision Central, set the following parameters:
   - **Username for the Maven service hosted by Decision Central** (**`DECISION_CENTRAL_MAVEN_USERNAME`**): The user name for the built-in Maven repository.
   - **Password for the Maven service hosted by Decision Central** (**`DECISION_CENTRAL_MAVEN_PASSWORD`**): The password for the built-in Maven repository.

5. You can use Git hooks to facilitate interaction between the internal Git repository of Decision Central and an external Git repository. To configure Git hooks, set the following parameter:

   - **Git hooks directory** (**`GIT_HOOKS_DIR`**): The fully qualified path to a Git hooks directory, for example, **`/opt/eap/standalone/data/kie/git/hooks`**. You must provide the content of this directory and mount it at the specified path; for instructions, see Section 3.3, “Providing the Git hooks directory”.

6. If you want to use RH-SSO or LDAP authentication, complete the following additional configuration. Do not configure LDAP authentication and RH-SSO authentication in the same deployment.

   a. In the RH-SSO or LDAP service, create all user names in the deployment parameters. If you do not set any of the parameters, create users with the default user names. The created users must also be assigned to roles:

      - **`KIE_ADMIN_USER`**: default user name **`adminUser`**, roles: **`kie-server,rest-all,admin`**
      - **`KIE_SERVER_CONTROLLER_USER`**: default user name **`controllerUser`**, roles: **`kie-server,rest-all,guest`**
      - **`DECISION_CENTRAL_MAVEN_USERNAME`** (not needed if you configure the use of an external Maven repository): default user name **`mavenUser`**. No roles are required.
      - **`KIE_SERVER_USER`**: default user name **`executionUser`**, roles **`kie-server,rest-all,guest`**

b. If you want to configure Red Hat Single Sign On (RH-SSO) authentication, an RH-SSO realm that applies to Red Hat Decision Manager must exist. Decision Server. If the client does not yet exist, the template can create it during deployment. Clients within RH-SSO must also exist for Decision Central and for Decision Server. If the clients do not yet exist, the template can create them during deployment.
For the user roles that you can configure in RH-SSO, see Roles and users.

Use one of the following procedures:

i. If the clients for Red Hat Decision Manager within RH-SSO already exist, set the following parameters in the template:

- **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.
- **Decision Central RH-SSO Client name** (**`DECISION_CENTRAL_SSO_CLIENT`**): The RH-SSO client name for Decision Central.
- **Decision Central RH-SSO Client Secret** (**`DECISION_CENTRAL_SSO_SECRET`**): The secret string that is set in RH-SSO for the client for Decision Central.
- **KIE Server RH-SSO Client name** (**`KIE_SERVER_SSO_CLIENT`**): The RH-SSO client name for Decision Server.
- **KIE Server RH-SSO Client Secret** (**`KIE_SERVER_SSO_SECRET`**): The secret string that is set in RH-SSO for the client for Decision Server.
- **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.

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

- **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.
- **Decision Central RH-SSO Client name** (**`DECISION_CENTRAL_SSO_CLIENT`**): The name of the client to create in RH-SSO for Decision Central.
- **Decision Central RH-SSO Client Secret** (**`DECISION_CENTRAL_SSO_SECRET`**): The secret string to set in RH-SSO for the client for Decision Central.
- **Decision Central Custom http Route Hostname** (**`DECISION_CENTRAL_HOSTNAME_HTTP`**): The fully qualified host name to use for the HTTP endpoint for Decision Central. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **Decision Central Custom https Route Hostname** (**`DECISION_CENTRAL_HOSTNAME_HTTPS`**): The fully qualified host name to use for the HTTPS endpoint for Decision Central. If you need to create a client in RH-SSO, you can not leave this parameter blank.

- **KIE Server RH-SSO Client name** (`KIE_SERVER_SSO_CLIENT`): The name of the client to create in RH-SSO for Decision Server.
- **KIE Server RH-SSO Client Secret** (`KIE_SERVER_SSO_SECRET`): The secret string to set in RH-SSO for the client for Decision Server.
- **KIE Server Custom http Route Hostname** (`KIE_SERVER_HOSTNAME_HTTP`): The fully qualified host name to use for the HTTP endpoint for Decision Server. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **KIE Server Custom https Route Hostname** (`KIE_SERVER_HOSTNAME_HTTPS`): The fully qualified host name to use for the HTTPS endpoint for Decision Server. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **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.
- **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.

c. To configure LDAP, set the `AUTH_LDAP*` parameters of the template. These parameters correspond to the settings of the LdatExtended 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 pathname of a file that defines role mapping, for example, `/opt/eap/standalone/configuration/rolemapping/rolemapping.properties`. You must provide this file and mount it at this path in all applicable deployment configurations; for instructions, see Section 3.5, “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`.

7. Complete the creation of the environment, depending on the method that you are using:
   - 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 and run the command line.

3.2. DEPLOYING HIGH-AVAILABILITY DECISION CENTRAL AND ONE DECISION SERVER IN AN AUTHORING OR MANAGED SERVER ENVIRONMENT

To deploy high-availability Decision Central and one Decision Server in an authoring or managed server environment, use the **`rhdm72-authoring-ha.yaml`** template file. You can download the file from the Software Downloads page.

IMPORTANT

In the current version, the high-availability functionality is a technology preview.

Procedure

1. Use one of the following methods to deploy the template:
   - In the OpenShift Web UI, select **Add to Project** → **Import YAML / JSON** and then select or paste the **`rhdm72-authoring-ha.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>/rhdm72-authoring-ha.yaml -p
     DECISION_CENTRAL_HTTPS_SECRET=decisioncentral-app-secret -p
     KIE_SERVER_HTTPS_SECRET=kieserver-app-secret
     ```

     In this command line:
     - 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. You can view the template file to see descriptions for all parameters.
2. Set the following parameters as necessary:
   - **Decision Central Server Keystore Secret Name** (**`DECISION_CENTRAL_HTTPS_SECRET`**): The name of the secret for Decision Central, as created in Section 2.3, “Creating the secrets for Decision Central”.
   - **KIE Server Keystore Secret Name** (**`KIE_SERVER_HTTPS_SECRET`**): The name of the secret for Decision Server, as created in Section 2.2, “Creating the secrets for Decision Server”.
   - **Application Name** (**`APPLICATION_NAME`**): The name of the OpenShift application. It is used in the default URLs for Decision Central and Decision 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) on the Decision Central that the Decision Server is to join.
   - **Decision Central Server Certificate Name** (**`DECISION_CENTRAL_HTTPS_NAME`**): The name of the certificate in the keystore that you created in Section 2.3, “Creating the secrets for Decision Central”.
   - **Decision Central Server Keystore Password** (**`DECISION_CENTRAL_HTTPS_PASSWORD`**): The password for the keystore that you created in Section 2.3, “Creating the secrets for Decision Central”.

- **KIE Server Certificate Name** (**`KIE_SERVER_HTTPS_NAME`**): The name of the certificate in the keystore that you created in Section 2.2, “Creating the secrets for Decision Server”.
- **KIE Server Keystore Password** (**`KIE_SERVER_HTTPS_PASSWORD`**): The password for the keystore that you created in Section 2.2, “Creating the secrets for Decision Server”.
- **ImageStream Namespace** (**`IMAGE_STREAM_NAMESPACE`**): The namespace where the image streams are available. If the image streams were already available in your OpenShift environment (see Section 2.1, “Ensuring the availability of image streams and the image registry”), the namespace is **`openshift`**. If you have installed the image streams file, the namespace is the name of the OpenShift project.
  You can also set the following user names and passwords:
- **KIE Admin User** (**`KIE_ADMIN_USER`**) and **KIE Admin Password** (**`KIE_ADMIN_PWD`**): The user name and password for the administrative user in Decision Central.
- **KIE Server User** (**`KIE_SERVER_USER`**) and **KIE Server Password** (**`KIE_SERVER_PWD`**): The user name and password that a client application must use to connect to the Decision Server.

3. If you want to deploy additional Decision Servers and connect them to this Decision Central, set the following parameters:
   - **KIE Server Controller User** (**`KIE_SERVER_CONTROLLER_USER`**) and **KIE Server Controller Password** (**`KIE_SERVER_CONTROLLER_PWD`**): The user name and password that a Decision Server must use to connect to the Decision Central.
4. If you want to place the built KJAR files into an external Maven repository, set the following parameters:
   - **Maven repository URL** (**`MAVEN_REPO_URL`**): The URL for the Maven repository.
   - **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.
   - **Maven repository ID** (**`MAVEN_REPO_ID`**): The Maven ID, which must match the **`id`** setting for the Maven repository.
     Alternatively, if you want to use the Maven repository that is built into Decision Central and to connect additional Decision Servers to the Decision Central, set the following parameters:
   - **Username for the Maven service hosted by Decision Central** (**`DECISION_CENTRAL_MAVEN_USERNAME`**): The user name for the built-in Maven repository.
   - **Password for the Maven service hosted by Decision Central** (**`DECISION_CENTRAL_MAVEN_PASSWORD`**): The password for the built-in Maven repository.
5. You can use Git hooks to facilitate interaction between the internal Git repository of Decision Central and an external Git repository. To configure Git hooks, set the following parameter:
   - **Git hooks directory** (**`GIT_HOOKS_DIR`**): The fully qualified path to a Git hooks directory, for example, **`/opt/eap/standalone/data/kie/git/hooks`**. You must provide the content of this directory and mount it at the specified path; for instructions, see Section 3.3,

"Providing the Git hooks directory".

6. If you want to use RH-SSO or LDAP authentication, complete the following additional configuration. Do not configure LDAP authentication and RH-SSO authentication in the same deployment.

   a. In the RH-SSO or LDAP service, create all user names in the deployment parameters. If you do not set any of the parameters, create users with the default user names. The created users must also be assigned to roles:

      - **`KIE_ADMIN_USER`**: default user name **`adminUser`**, roles: **`kie-server,rest-all,admin`**
      - **`KIE_SERVER_CONTROLLER_USER`**: default user name **`controllerUser`**, roles: **`kie-server,rest-all,guest`**
      - **`DECISION_CENTRAL_MAVEN_USERNAME`** (not needed if you configure the use of an external Maven repository): default user name **`mavenUser`**. No roles are required.
      - **`KIE_SERVER_USER`**: default user name **`executionUser`**, roles **`kie-server,rest-all,guest`**

   b. If you want to configure Red Hat Single Sign On (RH-SSO) authentication, an RH-SSO realm that applies to Red Hat Decision Manager must exist. Decision Server. If the client does not yet exist, the template can create it during deployment. Clients within RH-SSO must also exist for Decision Central and for Decision Server. If the clients do not yet exist, the template can create them during deployment.
   For the user roles that you can configure in RH-SSO, see Roles and users.

      Use one of the following procedures:

      i. If the clients for Red Hat Decision Manager within RH-SSO already exist, set the following parameters in the template:

         - **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.
         - **Decision Central RH-SSO Client name** (**`DECISION_CENTRAL_SSO_CLIENT`**): The RH-SSO client name for Decision Central.
         - **Decision Central RH-SSO Client Secret** (**`DECISION_CENTRAL_SSO_SECRET`**): The secret string that is set in RH-SSO for the client for Decision Central.
         - **KIE Server RH-SSO Client name** (**`KIE_SERVER_SSO_CLIENT`**): The RH-SSO client name for Decision Server.
         - **KIE Server RH-SSO Client Secret** (**`KIE_SERVER_SSO_SECRET`**): The secret string that is set in RH-SSO for the client for Decision Server.
         - **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.

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

- **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.
- **Decision Central RH-SSO Client name** (**`DECISION_CENTRAL_SSO_CLIENT`**): The name of the client to create in RH-SSO for Decision Central.
- **Decision Central RH-SSO Client Secret** (**`DECISION_CENTRAL_SSO_SECRET`**): The secret string to set in RH-SSO for the client for Decision Central.
- **Decision Central Custom http Route Hostname** (**`DECISION_CENTRAL_HOSTNAME_HTTP`**): The fully qualified host name to use for the HTTP endpoint for Decision Central. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **Decision Central Custom https Route Hostname** (**`DECISION_CENTRAL_HOSTNAME_HTTPS`**): The fully qualified host name to use for the HTTPS endpoint for Decision Central. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **KIE Server RH-SSO Client name** (**`KIE_SERVER_SSO_CLIENT`**): The name of the client to create in RH-SSO for Decision Server.
- **KIE Server RH-SSO Client Secret** (**`KIE_SERVER_SSO_SECRET`**): The secret string to set in RH-SSO for the client for Decision Server.
- **KIE Server Custom http Route Hostname** (**`KIE_SERVER_HOSTNAME_HTTP`**): The fully qualified host name to use for the HTTP endpoint for Decision Server. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **KIE Server Custom https Route Hostname** (**`KIE_SERVER_HOSTNAME_HTTPS`**): The fully qualified host name to use for the HTTPS endpoint for Decision Server. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **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.
- **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.

c. To configure LDAP, set the **`AUTH_LDAP*`** parameters of the template. These parameters correspond to the settings of the LdatExtended 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 pathname of a file that defines role mapping, for example, **`/opt/eap/standalone/configuration/rolemapping/rolemapping.properties`**. You must provide this file and mount it at this path in all applicable deployment configurations; for instructions, see Section 3.5, “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`**.

7. If an AMQ 7.1 image is not available in the **`openshift`** namespace with default settings, set the following parameters:
   - **AMQ ImageStream Namespace** (**`AMQ_IMAGE_STREAM_NAMESPACE`**): Namespace in which the ImageStream for the AMQ image is installed. The default setting is **`openshift`**.
   - **AMQ ImageStream Name** (**`AMQ_IMAGE_STREAM_NAME`**): The name of the image stream for the AMQ broker. The default setting is **`amq-broker71-openshift`**.
   - **AMQ ImageStream Tag** (**`AMQ_IMAGE_STREAM_TAG`**): The AMQ image stream tag. The default setting is **`1.0`**.
8. Complete the creation of the environment, depending on the method that you are using:
   - 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 and run the command line.

3.3. 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 Decision 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 pushing commits to an upstream repository is required, complete the following steps to enable access to the repository:
   a. Create a pair of a secret key named **`id_rsa`** and public key named **`id_rsa.pub`**. Use an email address that can access the upstream repository. Set an empty passphrase. For instructions, see Generating a new SSH key.
   b. Upload the public key to the upstream repository.
   c. Create an OpenShift secret named **`rhdm-centr-gitkey`** from the secret key:

```
oc create secret generic --from-file=id_rsa=id_rsa rhdm-centr-
gitkey
```

   d. Mount the secret in the ssh key path of the Decision Central deployment:

```
oc set volume dc/<myapp>-rhdmcentr --add --type secret --secret-
name rhdm-centr-gitkey --mount-path=/home/jboss/.ssh --name=ssh-
key
```

Where **<myapp>** is the application name that was 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 Decision Central from a repository, this repository remains configured as the upstream repository. Create a file named **post-commit** with permission values **755** and the following content:

```
git push
```

3. Supply the Git hooks directory to the Decision 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> ...
```

      Where **file_1**, **file_2** and so on are git hook script files. For example:

```
oc create configmap git_hooks --from-file=post-commit=post-
commit
```

      iii. Mount the configuration map on the Decision 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
```

      Where **<myapp>** is 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 (where *myapp* is 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 yor project. Because Decision Central does not start until you provide the Git hooks directory, the Decision 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 Decision Server pod is not present, start it:

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

Where **<myapp>** is the application name that was set when configuring the template.

3.4. DEPLOYING AN ADDITIONAL DECISION SERVER

As a part of a managed server infrastructure, you can deploy an additional Decision Server on the OpenShift infrastructure. You can then use Decision Central to deploy, undeploy, and manage services on this Decision Server.

To deploy an additional Decision Server, use the **rhdm72-kieserver.yaml** template file. You can download the file from the Software Downloads page.

Procedure

1. Use one of the following methods to deploy the template:
   - In the OpenShift Web UI, select **Add to Project** → **Import YAML / JSON** and then select or paste the **rhdm72-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>/rhdm72-kieserver.yaml -p
KIE_SERVER_HTTPS_SECRET=kieserver-app-secret
```

     In this command line:
     - 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. You can view the template file to see descriptions for all parameters.
2. Set the following parameters:
   - **KIE server controller service** (**KIE_SERVER_CONTROLLER_SERVICE**): The OpenShift service name for the Decision Central that you installed for this environment.
   - **KIE server controller user** (**KIE_SERVER_CONTROLLER_USER**): The controller user name for logging onto the Decision Central that you configured.
   - **KIE server controller password** (**KIE_SERVER_CONTROLLER_PWD**): The controller password for logging onto the Decision Central that you configured.
   - **KIE Server Keystore Secret Name** (**KIE_SERVER_HTTPS_SECRET**): The name of the secret for Decision Server, as created in Section 2.2, “Creating the secrets for Decision Server”.

- **Application Name** (**`APPLICATION_NAME`**): The name of the OpenShift application. It is used in the default URL for Decision 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) on the Decision Central that the Decision Server is to join.

- **KIE Server Certificate Name** (**`KIE_SERVER_HTTPS_NAME`**): The name of the certificate in the keystore that you created in Section 2.2, “Creating the secrets for Decision Server”.

- **KIE Server Keystore Password** (**`KIE_SERVER_HTTPS_PASSWORD`**): The password for the keystore that you created in Section 2.2, “Creating the secrets for Decision Server”.

3. Set the parameters for access to the Maven repository, depending on whether you configured the Decision Central to use the built-in or external repository:

   a. For a built-in repository:

      - **Name of the Maven service hosted by Decision Central** (**`DECISION_CENTRAL_MAVEN_SERVICE`**): The service name for the built-in Maven repository of the Decision Central.

      - **Username for the Maven service hosted by Decision Central** (**`DECISION_CENTRAL_MAVEN_USERNAME`**): The user name for the built-in Maven repository of the Decision Central. Enter the user name that you configured for the Decision Central as **`DECISION_CENTRAL_MAVEN_USERNAME`**.

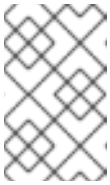
      - **Password to access the Maven service hosted by Decision Central** (**`DECISION_CENTRAL_MAVEN_PASSWORD`**): The password for the built-in Maven repository of the Decision Central. Enter the password that you configured for the Decision Central as **`DECISION_CENTRAL_MAVEN_PASSWORD`**.

   b. For an external repository:

      - **Maven repository URL** (**`MAVEN_REPO_URL`**): The URL for the Maven repository with services.

      - **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.



NOTE

You can set up access to both the built-in Maven repository of the Decision Central an external Maven repository if your services have dependencies in both repositories.

4. If you want to use RH-SSO or LDAP authentication, complete the following additional configuration. Do not configure LDAP authentication and RH-SSO authentication in the same deployment.

   a. In the RH-SSO or LDAP service, create all user names in the deployment parameters. If you do not set any of the parameters, create users with the default user names. The created users must also be assigned to roles:

users must also be assigned to roles.

- **KIE_ADMIN_USER**: default user name **adminUser**, roles: **kie-server,rest-all,admin**
- **KIE_SERVER_USER**: default user name **executionUser**, roles **kie-server,rest-all,guest**

b. If you want to configure Red Hat Single Sign On (RH-SSO) authentication, an RH-SSO realm that applies to Red Hat Decision Manager must exist. A client within RH-SSO must also exist for
For the user roles that you can configure in RH-SSO, see Roles and users.

Use one of the following procedures:

i. If the client for Red Hat Decision Manager within RH-SSO already exists, set the following parameters in the template:

- **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.
- **KIE Server RH-SSO Client name** (**KIE_SERVER_SSO_CLIENT**): The RH-SSO client name for Decision Server.
- **KIE Server RH-SSO Client Secret** (**KIE_SERVER_SSO_SECRET**): The secret string that is set in RH-SSO for the client for Decision Server.
- **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.

ii. To create the client for Red Hat Decision Manager within RH-SSO, set the following parameters in the template:

- **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.
- **KIE Server RH-SSO Client name** (**KIE_SERVER_SSO_CLIENT**): The name of the client to create in RH-SSO for Decision Server.
- **KIE Server RH-SSO Client Secret** (**KIE_SERVER_SSO_SECRET**): The secret string to set in RH-SSO for the client for Decision Server.
- **KIE Server Custom http Route Hostname** (**KIE_SERVER_HOSTNAME_HTTP**): The fully qualified host name to use for the HTTP endpoint for Decision Server. If you need to create a client in RH-SSO, you can not leave this parameter blank.
- **KIE Server Custom https Route Hostname** (**KIE_SERVER_HOSTNAME_HTTPS**): The fully qualified host name to use for the HTTPS endpoint for Decision Server. If you need to create a client in RH-SSO, you can not leave this parameter blank.

- **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.
- **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.

c. To configure LDAP, set the **AUTH_LDAP*** parameters of the template. These parameters correspond to the settings of the LdatExtended 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 pathname of a file that defines role mapping, for example, **/opt/eap/standalone/configuration/rolemapping/rolemapping.properties**. You must provide this file and mount it at this path in all applicable deployment configurations; for instructions, see Section 3.5, “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**.

5. Complete the creation of the environment, depending on the method that you are using:
   - In the OpenShift Web UI, click **Create**.
     - A **This will create resources that may have security or project behavior implications** pop-up message might be displayed. If it is displayed, click **Create Anyway**.
   - Complete and run the command line.

3.5. 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. Run the following command:

```
oc create configmap ldap_role_mapping --from-file=<new_name>=
<existing_name>
```

Where **new_name** is 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** is the name of the file that you created. For example:

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

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

   - ***myapp*-rhdmcentr**: Decision Central
   - ***myapp*-kieserver**: Decision Server

   Where **myapp** is the application name. Sometimes, several Decision 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
```

   Where **mapping_dir** is the directory name (without file name) set in the **AUTH_ROLE_MAPPER_ROLES_PROPERTIES** parameter, for example, **/opt/eap/standalone/configuration/rolemapping** .

CHAPTER 4. OPENSHIFT TEMPLATE REFERENCE INFORMATION

Red Hat Decision Manager provides the following OpenShift templates. To access the templates, download and extract the **`rhdm-7.2.0-openshift-templates.zip`** product deliverable file from the Software Downloads page of the Red Hat customer portal.

- **`rhdm72-authoring.yaml`** provides a Decision Central and a Decision Server connected to the Decision 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 4.1, “rhdm72-authoring.yaml template”.
- **`rhdm72-authoring-ha.yaml`** provides a high-availability Decision Central and a Decision Server connected to the Decision Central. You can use this environment to author services and other business assets or to run them in staging or production environments. The high-availability functionality is in technical preview. For details about this template, see Section 4.2, “rhdm72-authoring-ha.yaml template”.
- **`rhdm72-kieserver.yaml`** provides a Decision Server. You can configure the Decision Server to connect to a Decision Central. In this way, you can set up a staging or production environment in which one Decision Central manages several distinct Decision Servers. For details about this template, see Section 4.3, “rhdm72-kieserver.yaml template”.

4.1. RHDM72-AUTHORING.YAML TEMPLATE

Application template for a non-HA persistent authoring environment, for Red Hat Decision Manager 7.2

4.1.1. Parameters

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

Variable name	Image Environment Variable	Description	Example value	Required
APPLICATION_NAME	—	The name for the application.	myapp	True
KIE_ADMIN_USER	**KIE_ADMIN_USER**	KIE administrator username	adminUser	False
KIE_ADMIN_PWD	**KIE_ADMIN_PWD**	KIE administrator password	—	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_CONTROLLER_USER	**KIE_SERVER_CONTROLLER_USER**	KIE server controller username (Sets the org.kie.server.controller.user system property)	controllerUser	False
KIE_SERVER_CONTROLLER_PWD	**KIE_SERVER_CONTROLLER_PWD**	KIE server controller password (Sets the org.kie.server.controller.pwd system property)	—	False
KIE_SERVER_CONTROLLER_TOKEN	**KIE_SERVER_CONTROLLER_TOKEN**	KIE server controller token for bearer authentication (Sets the org.kie.server.controller.token system property)	—	False
KIE_SERVER_USER	**KIE_SERVER_USER**	KIE server username (Sets the org.kie.server.user system property)	executionUser	False
KIE_SERVER_PWD	**KIE_SERVER_PWD**	KIE server password (Sets the org.kie.server.pwd system property)	—	False
KIE_SERVER_BYPASS_AUTH_USER	**KIE_SERVER_BYPASS_AUTH_USER**	KIE server bypass auth user (Sets the org.kie.server.bypass.auth.user system property)	false	False

Variable name	Image Environment Variable	Description	Example value	Required
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
DECISION_CENTRAL_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<project>.<default-domain-suffix>	—	False
DECISION_CENTRAL_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for https service route. Leave blank for default hostname, e.g.: secure-<application-name>-rhdmcentr-<project>.<default-domain-suffix>	—	False
KIE_SERVER_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route. Leave blank for default hostname, e.g.: <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. Leave blank for default hostname, e.g.: secure-<application-name>-kieserver-<project>.<default-domain-suffix>	—	False
KIE_SERVER_USE_SECURE_ROUTE_NAME	**KIE_SERVER_USE_SECURE_ROUTE_NAME**	If true, will use secure-APPLICATION_NAME-kieserver vs. APPLICATION_NAME-kieserver as the route name.	false	False
DECISION_CENTRAL_HTTPS_SECRET	—	The name of the secret containing the keystore file	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

Variable name	Image Environment Variable	Description	Example value	Required
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
IMAGE_STREAM_NAMESPACE	—	Namespace in which the ImageStreams for Red Hat Middleware images are installed. These ImageStreams are normally installed in the openshift namespace. You should only need to modify this if you installed the ImageStreams in a different namespace/project.	openshift	True
KIE_SERVER_IMAGE_STREAM_NAME	—	The name of the image stream to use for KIE server. Default is "rhdm72-kieserver-openshift".	rhdm72-kieserver-openshift	True
IMAGE_STREAM_TAG	—	A named pointer to an image in an image stream. Default is "1.1".	1.1	True
MAVEN_REPO_ID	**MAVEN_REPO_ID**	The id to use for the maven repository, if set. Default is generated randomly.	my-repo-id	False

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_REPO_URL	**MAVEN_REPO_URL**	Fully qualified URL to a Maven repository or service.	http://nexus.nexus-project.svc.cluster.local:8081/nexus/content/groups/public/	False
MAVEN_REPO_USERNAME	**MAVEN_REPO_USERNAME**	Username to access the Maven repository, if required.	—	False
MAVEN_REPO_PASSWORD	**MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	—	False
DECISION_CENTRAL_MAVEN_USERNAME	**KIE_MAVEN_USER**	Username to access the Maven service hosted by Decision Central inside EAP.	mavenUser	True
DECISION_CENTRAL_MAVEN_PASSWORD	**KIE_MAVEN_PWD**	Password to access the Maven service hosted by Decision Central inside EAP.	—	True
GIT_HOOKS_DIR	**GIT_HOOKS_DIR**	The directory to use for git hooks, if required.	**/opt/eap/standalone/data/kie/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	2Gi	False
KIE_SERVER_MEMORY_LIMIT	—	KIE server Container memory limit	1Gi	False

Variable name	Image Environment Variable	Description	Example value	Required
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 Username used to create the Client if it doesn't exist	—	False
SSO_PASSWORD	**SSO_PASSWORD**	RH-SSO Realm Admin Password used to create the Client	—	False
SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	**SSO_DISABLE_SSL_CERTIFICATE_VALIDATION**	RH-SSO Disable SSL Certificate Validation	false	False
SSO_PRINCIPAL_ATTRIBUTE	**SSO_PRINCIPAL_ATTRIBUTE**	RH-SSO Principal Attribute to use as username.	preferred_username	False
AUTH_LDAP_URL	**AUTH_LDAP_URL**	LDAP Endpoint to connect for authentication	ldap://myldap.example.com	False

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

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

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

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

4.1.2. 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.

4.1.2.1. Services

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

Service	Port	Name	Description
${APPLICATION_NAME}-rhdmcentr	8080	http	All the Decision Central web server's ports.
	8443	https	
	8001	git-ssh	
${APPLICATION_NAME}-kieserver	8080	http	All the KIE server web server's ports.
	8443	https	

4.1.2.2. Routes

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

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

4.1.2.3. Deployment Configurations

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

4.1.2.3.1. Triggers

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

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

4.1.2.3.2. Replicas

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

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

4.1.2.3.3. Pod Template

4.1.2.3.3.1. Service Accounts

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

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

4.1.2.3.3.2. Image

Deployment	Image
${APPLICATION_NAME}-rhdmcentr	rhdm72-decisioncentral-openshift
${APPLICATION_NAME}-kieserver	**${KIE_SERVER_IMAGE_STREAM_NAME}**

4.1.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-rhdmcentr

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/kie-drools-wb.jsp
```

${APPLICATION_NAME}-kieserver

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/services/rest/server/readycheck
```

4.1.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-rhdmcentr

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/kie-drools-wb.jsp
```

${APPLICATION_NAME}-kieserver

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/services/rest/server/readycheck
```

4.1.2.3.3.5. Exposed Ports

Deployments	Name	Port	Protocol
${APPLICATION_NAME}-rhdmcentr	jolokia	8778	**TCP**
	http	8080	**TCP**
	https	8443	**TCP**
	git-ssh	8001	**TCP**
${APPLICATION_NAME}-kieserver	jolokia	8778	**TCP**
	http	8080	**TCP**
	https	8443	**TCP**

4.1.2.3.3.6. Image Environment Variables

Deployment	Variable name	Description	Example value
${APPLICATION_NAME}-rhdmcentr	**KIE_ADMIN_USER**	KIE administrator username	**${KIE_ADMIN_USER}**
	KIE_ADMIN_PWD	KIE administrator password	**${KIE_ADMIN_PWD}**

Deployment	Variable name	Description	Example value
	KIE_MBEANS	KIE server mbeans enabled/disabled (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_CONTROLLER_USER	KIE server controller username (Sets the org.kie.server.controller.user system property)	**${KIE_SERVER_CONTROLLER_USER}**
	KIE_SERVER_CONTROLLER_PWD	KIE server controller password (Sets the org.kie.server.controller.pwd system property)	**${KIE_SERVER_CONTROLLER_PWD}**
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	KIE_SERVER_USER	KIE server username (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**
	KIE_SERVER_PWD	KIE server password (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	WORKBENCH_ROUTE_NAME	—	**${APPLICATION_NAME}-rhdmcentr**
	MAVEN_REPO_ID	The id to use for the maven repository, if set. Default is generated randomly.	**${MAVEN_REPO_ID}**
	MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**

Deployment	Variable name	Description	Example value
	MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	KIE_MAVEN_USER	Username to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_USERNAME}**
	KIE_MAVEN_PWD	Password to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_PASSWORD}**
	GIT_HOOKS_DIR	The directory to use for git hooks, if required.	**${GIT_HOOKS_DIR}**
	HTTPS_KEYSTORE_DIR	—	**/etc/decisioncentral-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret	**${DECISION_CENTRAL_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate	**${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}**

Deployment	Variable name	Description	Example value
	SSO_USERNAME	RH-SSO Realm Admin Username used to create the Client if it doesn't exist	**${SSO_USERNAME}**
	SSO_PASSWORD	RH-SSO Realm Admin Password used to create the Client	**${SSO_PASSWORD}**
	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO Disable SSL Certificate Validation	**${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}**
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as username.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<project>.<default-domain-suffix>	**${DECISION_CENTRAL_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for https service route. Leave blank for default hostname, e.g.: secure-<application-name>-rhdmcentr-<project>.<default-domain-suffix>	**${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}**

Deployment	Variable name	Description	Example value
	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}**
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	**${AUTH_LDAP_PARSE_USERNAME}**

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

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

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

Deployment	Variable name	Description	Example value
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This 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}**
${APPLICATION_NAME}-kieserver	**DROOLS_SERVER_FILTER_CLASSES**	KIE server class filtering (Sets the org.drools.server.filter.classes system property)	**${DROOLS_SERVER_FILTER_CLASSES}**
	KIE_ADMIN_USER	KIE administrator username	**${KIE_ADMIN_USER}**
	KIE_ADMIN_PWD	KIE administrator password	**${KIE_ADMIN_PWD}**
	KIE_MBEANS	KIE server mbeans enabled/disabled (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_BYPASS_AUTH_USER	KIE server bypass auth user (Sets the org.kie.server.bypass.auth.user system property)	**${KIE_SERVER_BYPASS_AUTH_USER}**
	KIE_SERVER_CONTROLLER_USER	KIE server controller username (Sets the org.kie.server.controller.user system property)	**${KIE_SERVER_CONTROLLER_USER}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_CONTROLLER_PWD	KIE server controller password (Sets the org.kie.server.controller.pwd system property)	**${KIE_SERVER_CONTROLLER_PWD}**
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	KIE_SERVER_CONTROLLER_SERVICE	—	**${APPLICATION_NAME}-rhdmcentr**
	KIE_SERVER_CONTROLLER_PROTOCOL	—	ws
	KIE_SERVER_ID	—	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_ROUTE_NAME	—	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_USE_SECURE_ROUTE_NAME	If true, will use secure-APPLICATION_NAME-kieserver vs. APPLICATION_NAME-kieserver as the route name.	**${KIE_SERVER_USE_SECURE_ROUTE_NAME}**
	KIE_SERVER_USER	KIE server username (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**
	KIE_SERVER_PWD	KIE server password (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	MAVEN_REPOS	—	RHDMCENTR,EXTERNAL
	RHDMCENTR_MAVEN_REPO_SERVICE	—	**${APPLICATION_NAME}-rhdmcentr**

Deployment	Variable name	Description	Example value
	RHDMCENTR_MAVEN_REPO_PATH	—	**/maven2/**
	RHDMCENTR_MAVEN_REPO_USERNAME	Username to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_USERNAME}**
	RHDMCENTR_MAVEN_REPO_PASSWORD	Password to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_PASSWORD}**
	EXTERNAL_MAVEN_REPO_ID	The id to use for the maven repository, if set. Default is generated randomly.	**${MAVEN_REPO_ID}**
	EXTERNAL_MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	EXTERNAL_MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	EXTERNAL_MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	HTTPS_KEYSTORE_DIR	—	**/etc/kieserver-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret	**${KIE_SERVER_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate	**${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

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

Deployment	Variable name	Description	Example value
	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}**
	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}**

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

4.1.2.3.3.7. 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

4.1.2.4. External Dependencies

4.1.2.4.1. Volume Claims

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

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

4.1.2.4.2. Secrets

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

decisioncentral-app-secret kieserver-app-secret

4.2. RHDM72-AUTHORING-HA.YAML TEMPLATE

Application template for a HA persistent authoring environment, for Red Hat Decision Manager 7.2

4.2.1. Parameters

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

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

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_CONTROLLER_TOKEN	**KIE_SERVER_CONTROLLER_TOKEN**	KIE server controller token for bearer authentication (Sets the org.kie.server.controller.token system property)	—	False
KIE_SERVER_USER	**KIE_SERVER_USER**	KIE server username (Sets the org.kie.server.user system property)	executionUser	False
KIE_SERVER_PWD	**KIE_SERVER_PWD**	KIE server password (Sets the org.kie.server.pwd system property)	—	False
KIE_SERVER_BYPASS_AUTH_USER	**KIE_SERVER_BYPASS_AUTH_USER**	KIE server bypass auth user (Sets the org.kie.server.bypass.auth.user system property)	false	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
DECISION_CENTRAL_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<project>.<default-domain-suffix>	—	False
DECISION_CENTRAL_HOSTNAME_HTTPS	**HOSTNAME_HTTPS**	Custom hostname for https service route. Leave blank for default hostname, e.g.: secure-<application-name>-rhdmcentr-<project>.<default-domain-suffix>	—	False
KIE_SERVER_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route. Leave blank for default hostname, e.g.: <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.: secure-<application-name>-kieserver-<project>.<default-domain-suffix>	—	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_USE_SECURE_ROUTE_NAME	**KIE_SERVER_USE_SECURE_ROUTE_NAME**	If true, will use secure-APPLICATION_NAME-kieserver vs. APPLICATION_NAME-kieserver as the route name.	false	False
DECISION_CENTRAL_HTTPS_SECRET	—	The name of the secret containing the keystore file	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
KIE_SERVER_HTTPS_PASSWORD	**HTTPS_PASSWORD**	The password for the keystore and certificate	mykeystorepass	False

Variable name	Image Environment Variable	Description	Example value	Required
APPFORMER_ELASTIC_RETRIES	**APPFORMER_ELASTIC_RETRIES**	The number of times that appformer will try to connect to the elasticsearch node before give up.	—	False
APPFORMER_JMS_BROKER_PORT	**APPFORMER_JMS_BROKER_PORT**	The port to connect to the JMS broker. Defaults to 61616	—	False
APPFORMER_JMS_BROKER_USER	**APPFORMER_JMS_BROKER_USER**	The username to connect to the JMS broker.	jmsBrokerUser	True
APPFORMER_JMS_BROKER_PASSWORD	**APPFORMER_JMS_BROKER_PASSWORD**	The password to connect to the JMS broker.	—	True
ES_HOSTNAME_HTTP	—	Custom hostname for http service route. Leave blank for default hostname, e.g.: <application-name>-rhdmindex-<project>.<default-domain-suffix>	—	False
APPFORMER_ELASTIC_CLUSTER_NAME	**APPFORMER_ELASTIC_CLUSTER_NAME**	Sets the ES cluster.name and configure it on Decision Central. Defaults to kie-cluster.	—	False
ES_NODE_NAME	**ES_NODE_NAME**	Sets the ES node.name property. Defaults to HOSTNAME env value.	—	False

Variable name	Image Environment Variable	Description	Example value	Required
ES_TRANSPORT_HOST	**ES_TRANSPORT_HOST**	Sets the ES transport.host property. This will set the transport address of the main ES cluster node. Used for communication between nodes in the cluster. Defaults to container address.	—	False
APPFORMER_ELASTIC_PORT	**APPFORMER_ELASTIC_PORT**	Sets the ES http.host property. This will set the http address of the main ES cluster node. Used for communication between nodes in the cluster and for communication with Decision Central.	—	False
ES_HTTP_HOST	**ES_HTTP_HOST**	Sets the ES http.host property. This will set the http address of the main ES cluster node. Used to interact with the cluster REST API. Defaults to the container IP address.	—	False
ES_HTTP_PORT	**ES_HTTP_PORT**	Sets the ES http.port property. This will set the http port of the main ES cluster node. Used to interact with cluster rest api.	—	False

Variable name	Image Environment Variable	Description	Example value	Required
ES_JAVA_OPTS	**ES_JAVA_OPTS**	Appends custom jvm configurations/properties to ES jvm.options configuration file.	-Xms1024m -Xmx1024m	False
AMQ_IMAGE_STREAM_NAMESPACE	—	Namespace in which the ImageStream for the AMQ image is installed. Default is "openshift".	openshift	True
AMQ_IMAGE_STREAM_NAME	—	The name of the image stream to use for the AMQ broker. Default is "amq-broker72-openshift".	amq-broker72-openshift	True
AMQ_IMAGE_STREAM_TAG	—	The AMQ image stream tag. Default is "1.1".	1.1	True
AMQ_ROLE	**AMQ_ROLE**	User role for standard broker user.	admin	True
AMQ_NAME	**AMQ_NAME**	The name of the broker	broker	True
AMQ_GLOBAL_MAX_SIZE	**AMQ_GLOBAL_MAX_SIZE**	Maximum amount of memory which message data may consume (Default: Undefined, half of the system's memory).	100 gb	False
ES_VOLUME_CAPACITY	—	Size of persistent storage for Elasticsearch volume.	1Gi	True

Variable name	Image Environment Variable	Description	Example value	Required
IMAGE_STREAM_NAMESPACE	—	Namespace in which the ImageStreams for Red Hat Middleware images are installed. These ImageStreams are normally installed in the openshift namespace. You should only need to modify this if you installed the ImageStreams in a different namespace/project.	openshift	True
KIE_SERVER_IMAGE_STREAM_NAME	—	The name of the image stream to use for KIE server. Default is "rhdm72-kieserver-openshift".	rhdm72-kieserver-openshift	True
IMAGE_STREAM_TAG	—	A named pointer to an image in an image stream. Default is "1.1".	1.1	True
MAVEN_REPO_ID	**MAVEN_REPO_ID**	The id to use for the maven repository, if set. Default is generated randomly.	my-repo-id	False
MAVEN_REPO_URL	**MAVEN_REPO_URL**	Fully qualified URL to a Maven repository or service.	http://nexus.nexus-project.svc.cluster.local:8081/nexus/content/groups/public/	False
MAVEN_REPO_USERNAME	**MAVEN_REPO_USERNAME**	Username to access the Maven repository, if required.	—	False

Variable name	Image Environment Variable	Description	Example value	Required
MAVEN_REPO_PASSWORD	**MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	—	False
DECISION_CENTRAL_MAVEN_USERNAME	**KIE_MAVEN_USER**	Username to access the Maven service hosted by Decision Central inside EAP.	mavenUser	True
DECISION_CENTRAL_MAVEN_PASSWORD	**KIE_MAVEN_PWD**	Password to access the Maven service hosted by Decision Central inside EAP.	—	True
GIT_HOOKS_DIR	**GIT_HOOKS_DIR**	The directory to use for git hooks, if required.	**/opt/eap/standalone/data/kie/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	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

Variable name	Image Environment Variable	Description	Example value	Required
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 Username used to create the Client if it doesn't exist	—	False
SSO_PASSWORD	**SSO_PASSWORD**	RH-SSO Realm Admin Password used to create the Client	—	False
SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	**SSO_DISABLE_SSL_CERTIFICATE_VALIDATION**	RH-SSO Disable SSL Certificate Validation	false	False
SSO_PRINCIPAL_ATTRIBUTE	**SSO_PRINCIPAL_ATTRIBUTE**	RH-SSO Principal Attribute to use as username.	preferred_username	False
AUTH_LDAP_URL	**AUTH_LDAP_URL**	LDAP Endpoint to connect for authentication	ldap://myldap.example.com	False
AUTH_LDAP_BIND_DN	**AUTH_LDAP_BIND_DN**	Bind DN used for authentication	uid=admin,ou=users,ou=exmample,ou=com	False
AUTH_LDAP_BIND_CREDENTIAL	**AUTH_LDAP_BIND_CREDENTIAL**	LDAP Credentials used for authentication	Password	False

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

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

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

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

4.2.2. 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.

4.2.2.1. Services

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

Service	Port	Name	Description
${APPLICATION_NAME}-rhdmcentr	8080	http	All the Decision Central web server's ports.
	8443	https	
	8001	git-ssh	
${APPLICATION_NAME}-ping	8888	ping	The JGroups ping port for clustering.
${APPLICATION_NAME}-kieserver	8080	http	All the KIE server web server's ports.
	8443	https	
${APPLICATION_NAME}-rhdmindex	9200	rest	All the Decision Central Indexing Elasticsearch ports.
	9300	transport	
${APPLICATION_NAME}-amq-tcp	61616	—	The broker's OpenWire port.

4.2.2.2. Routes

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

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

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

4.2.2.3. Deployment Configurations

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

4.2.2.3.1. Triggers

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

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

4.2.2.3.2. Replicas

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

Deployment	Replicas
${APPLICATION_NAME}-rhdmcentr	2
${APPLICATION_NAME}-kieserver	2
${APPLICATION_NAME}-rhdmindex	1
${APPLICATION_NAME}-amq	1

4.2.2.3.3. Pod Template

4.2.2.3.3.1. Service Accounts

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

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

4.2.2.3.3.2. Image

Deployment	Image
${APPLICATION_NAME}-rhdmcentr	rhdm72-decisioncentral-openshift
${APPLICATION_NAME}-kieserver	**${KIE_SERVER_IMAGE_STREAM_NAME}**
${APPLICATION_NAME}-rhdmindex	rhdm72-decisioncentral-indexing-openshift
${APPLICATION_NAME}-amq	**${AMQ_IMAGE_STREAM_NAME}**

4.2.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-rhdmcentr

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/kie-drools-wb.jsp
```

${APPLICATION_NAME}-kieserver

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/services/rest/server/readycheck
```

${APPLICATION_NAME}-rhdmindex

```
Http Get on http://localhost:9200/_cluster/health
```

${APPLICATION_NAME}-amq

```
/bin/bash -c /opt/amq/bin/readinessProbe.sh
```

4.2.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-rhdmcentr

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
```

```
http://localhost:8080/kie-drools-wb.jsp
```

${APPLICATION_NAME}-kieserver

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/services/rest/server/readycheck
```

4.2.2.3.3.5. 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**
${APPLICATION_NAME}-rhdmindex	es	9300	**TCP**
	http	9200	**TCP**
${APPLICATION_NAME}-amq	jolokia	8161	**TCP**
	amqp	5672	**TCP**
	mqtt	1883	**TCP**
	stomp	61613	**TCP**
	artemis	61616	**TCP**

4.2.2.3.3.6. Image Environment Variables

Deployment	Variable name	Description	Example value
${APPLICATION_NAME}-rhdmcentr	**KIE_ADMIN_USER**	KIE administrator username	**${KIE_ADMIN_USER}**

Deployment	Variable name	Description	Example value
	KIE_ADMIN_PWD	KIE administrator password	**${KIE_ADMIN_PWD}**
	KIE_MBEANS	KIE server mbeans enabled/disabled (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_CONTROLLER_USER	KIE server controller username (Sets the org.kie.server.controller.user system property)	**${KIE_SERVER_CONTROLLER_USER}**
	KIE_SERVER_CONTROLLER_PWD	KIE server controller password (Sets the org.kie.server.controller.pwd system property)	**${KIE_SERVER_CONTROLLER_PWD}**
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	KIE_SERVER_USER	KIE server username (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**
	KIE_SERVER_PWD	KIE server password (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	WORKBENCH_ROUTE_NAME	—	**${APPLICATION_NAME}-rhdmcentr**
	MAVEN_REPO_ID	The id to use for the maven repository, if set. Default is generated randomly.	**${MAVEN_REPO_ID}**
	MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**

Deployment	Variable name	Description	Example value
	MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	KIE_MAVEN_USER	Username to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_USERNAME}**
	KIE_MAVEN_PWD	Password to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_PASSWORD}**
	GIT_HOOKS_DIR	The directory to use for git hooks, if required.	**${GIT_HOOKS_DIR}**
	HTTPS_KEYSTORE_DIR	—	**/etc/decisioncentral-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret	**${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}**
	JGROUPS_PING_PROTOCOL	—	openshift.DNS_PING
	OPENSHIFT_DNS_PING_SERVICE_NAME	—	**${APPLICATION_NAME}-ping**
	OPENSHIFT_DNS_PING_SERVICE_PORT	—	8888

Deployment	Variable name	Description	Example value
	APPFORMER_ELASTIC_PORT	Sets the ES http.host property. This will set the http address of the main ES cluster node. Used for communication between nodes in the cluster and for communication with Decision Central.	**${APPFORMER_ELASTIC_PORT}**
	APPFORMER_ELASTIC_CLUSTER_NAME	Sets the ES cluster.name and configure it on Decision Central. Defaults to kie-cluster.	**${APPFORMER_ELASTIC_CLUSTER_NAME}**
	APPFORMER_ELASTIC_RETRIES	The number of times that appformer will try to connect to the elasticsearch node before give up.	**${APPFORMER_ELASTIC_RETRIES}**
	APPFORMER_ELASTIC_HOST	—	**${APPLICATION_NAME}-rhdmindex**
	APPFORMER_JMS_BROKER_ADDRESS	—	**${APPLICATION_NAME}-amq-tcp**
	APPFORMER_JMS_BROKER_PORT	The port to connect to the JMS broker. Defaults to 61616	**${APPFORMER_JMS_BROKER_PORT}**
	APPFORMER_JMS_BROKER_USER	The username to connect to the JMS broker.	**${APPFORMER_JMS_BROKER_USER}**
	APPFORMER_JMS_BROKER_PASSWORD	The password to connect to the JMS broker.	**${APPFORMER_JMS_BROKER_PASSWORD}**
	SSO_URL	RH-SSO URL	**${SSO_URL}**
	SSO_OPENIDCONNECT_DEPLOYMENTS	—	ROOT.war
	SSO_REALM	RH-SSO Realm name	**${SSO_REALM}**
	SSO_SECRET	Decision Central RH-SSO Client Secret	**${DECISION_CENTRAL_SSO_SECRET}**

Deployment	Variable name	Description	Example value
	SSO_CLIENT	Decision Central RH-SSO Client name	**${DECISION_CENTRAL_SSO_CLIENT}**
	SSO_USERNAME	RH-SSO Realm Admin Username used to create the Client if it doesn't exist	**${SSO_USERNAME}**
	SSO_PASSWORD	RH-SSO Realm Admin Password used to create the Client	**${SSO_PASSWORD}**
	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO Disable SSL Certificate Validation	**${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}**
	SSO_PRINCIPAL_ATTRIBUTE	RH-SSO Principal Attribute to use as username.	**${SSO_PRINCIPAL_ATTRIBUTE}**
	HOSTNAME_HTTP	Custom hostname for http service route. Leave blank for default hostname, e.g.: <application-name>-rhdmcentr-<project>.<default-domain-suffix>	**${DECISION_CENTRAL_HOSTNAME_HTTP}**
	HOSTNAME_HTTPS	Custom hostname for https service route. Leave blank for default hostname, e.g.: secure-<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}**

Deployment	Variable name	Description	Example value
	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.	**${AUTH_LDAP_SEARCH_TIME_LIMIT}**
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**

Deployment	Variable name	Description	Example value
	AUTH_LDAP_PARSE_USERNAME	A flag indicating if the DN is to be parsed for the username. If set to true, the DN is parsed for the username. If set to false the DN is not parsed for the username. This option is used together with usernameBeginString and usernameEndString.	**${AUTH_LDAP_PARSE_USERNAME}**
	AUTH_LDAP_USERNAME_BEGIN_STRING	Defines the String which is to be removed from the start of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_BEGIN_STRING}**
	AUTH_LDAP_USERNAME_END_STRING	Defines the String which is to be removed from the end of the DN to reveal the username. This option is used together with usernameEndString and only taken into account if parseUsername is set to true.	**${AUTH_LDAP_USERNAME_END_STRING}**
	AUTH_LDAP_ROLE_ATTRIBUTE_ID	Name of the attribute containing the user roles.	**${AUTH_LDAP_ROLE_ATTRIBUTE_ID}**
	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}**
${APPLICATION_NAME}-kieserver	**DROOLS_SERVER_FILTER_CLASSES**	KIE server class filtering (Sets the org.drools.server.filter.classes system property)	**${DROOLS_SERVER_FILTER_CLASSES}**
	KIE_ADMIN_PWD	KIE administrator password	**${KIE_ADMIN_PWD}**
	KIE_ADMIN_USER	KIE administrator username	**${KIE_ADMIN_USER}**
	KIE_MBEANS	KIE server mbeans enabled/disabled (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_BYPASS_AUTH_USER	KIE server bypass auth user (Sets the org.kie.server.bypass.auth.user system property)	**${KIE_SERVER_BYPASS_AUTH_USER}**
	KIE_SERVER_CONTROLLER_USER	KIE server controller username (Sets the org.kie.server.controller.user system property)	**${KIE_SERVER_CONTROLLER_USER}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_CONTROLLER_PWD	KIE server controller password (Sets the org.kie.server.controller.pwd system property)	**${KIE_SERVER_CONTROLLER_PWD}**
	KIE_SERVER_CONTROLLER_TOKEN	KIE server controller token for bearer authentication (Sets the org.kie.server.controller.token system property)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	KIE_SERVER_CONTROLLER_SERVICE	—	**${APPLICATION_NAME}-rhdmcentr**
	KIE_SERVER_CONTROLLER_PROTOCOL	—	ws
	KIE_SERVER_ID	—	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_ROUTE_NAME	—	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_USE_SECURE_ROUTE_NAME	If true, will use secure-APPLICATION_NAME-kieserver vs. APPLICATION_NAME-kieserver as the route name.	**${KIE_SERVER_USE_SECURE_ROUTE_NAME}**
	KIE_SERVER_PWD	KIE server password (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	KIE_SERVER_USER	KIE server username (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**
	MAVEN_REPOS	—	RHDMCENTR,EXTERNAL
	RHDMCENTR_MAVEN_REPO_SERVICE	—	**${APPLICATION_NAME}-rhdmcentr**
	RHDMCENTR_MAVEN_REPO_PATH	—	**/maven2/**

Deployment	Variable name	Description	Example value
	RHDMCENTR_MAVEN_REPO_USERNAME	Username to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_USERNAME}**
	RHDMCENTR_MAVEN_REPO_PASSWORD	Password to access the Maven service hosted by Decision Central inside EAP.	**${DECISION_CENTRAL_MAVEN_PASSWORD}**
	EXTERNAL_MAVEN_REPO_ID	The id to use for the maven repository, if set. Default is generated randomly.	**${MAVEN_REPO_ID}**
	EXTERNAL_MAVEN_REPO_URL	Fully qualified URL to a Maven repository or service.	**${MAVEN_REPO_URL}**
	EXTERNAL_MAVEN_REPO_USERNAME	Username to access the Maven repository, if required.	**${MAVEN_REPO_USERNAME}**
	EXTERNAL_MAVEN_REPO_PASSWORD	Password to access the Maven repository, if required.	**${MAVEN_REPO_PASSWORD}**
	HTTPS_KEYSTORE_DIR	—	**/etc/kieserver-secret-volume**
	HTTPS_KEYSTORE	The name of the keystore file within the secret	**${KIE_SERVER_HTTPS_KEYSTORE}**
	HTTPS_NAME	The name associated with the server certificate	**${KIE_SERVER_HTTPS_NAME}**
	HTTPS_PASSWORD	The password for the keystore and certificate	**${KIE_SERVER_HTTPS_PASSWORD}**
	SSO_URL	RH-SSO URL	**${SSO_URL}**

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

Deployment	Variable name	Description	Example value
	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.	**${AUTH_LDAP_SEARCH_TIME_LIMIT}**
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	The name of the attribute in the user entry that contains the DN of the user. This may be necessary if the DN of the user itself contains special characters, backslash for example, that prevent correct user mapping. If the attribute does not exist, the entry's DN is used.	**${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}**

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

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

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

Deployment	Variable name	Description	Example value
	AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	If you are not using referrals, you can ignore this option. When using referrals, this option denotes the attribute name which contains users defined for a certain role, for example member, if the role object is inside the referral. Users are checked against the content of this attribute name. If this option is not set, the check will always fail, so role objects cannot be stored in a referral tree.	**${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}**
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	When present, the RoleMapping Login Module will be configured to use the provided file. This property defines the fully-qualified file path and name of a properties file or resource which maps roles to replacement roles. The format is original_role=role1,role2, role3	**${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}**
${APPLICATION_NAME}-rhdmindex	**ES_CLUSTER_NAME**	Sets the ES cluster.name and configure it on Decision Central. Defaults to kie-cluster.	**${APPFORMER_ELASTIC_CLUSTER_NAME}**
	ES_NODE_NAME	Sets the ES node.name property. Defaults to HOSTNAME env value.	**${ES_NODE_NAME}**

Deployment	Variable name	Description	Example value
	ES_TRANSPORT_HOST	Sets the ES transport.host property. This will set the transport address of the main ES cluster node. Used for communication between nodes in the cluster. Defaults to container address.	**${ES_TRANSPORT_HOST}**
	ES_TRANSPORT_TCP_PORT	Sets the ES http.host property. This will set the http address of the main ES cluster node. Used for communication between nodes in the cluster and for communication with Decision Central.	**${APPFORMER_ELASTIC_PORT}**
	ES_HTTP_PORT	Sets the ES http.port property. This will set the http port of the main ES cluster node. Used to interact with cluster rest api.	**${ES_HTTP_PORT}**
	ES_HTTP_HOST	Sets the ES http.host property. This will set the http address of the main ES cluster node. Used to interact with the cluster REST API. Defaults to the container IP address.	**${ES_HTTP_HOST}**
	ES_JAVA_OPTS	Appends custom jvm configurations/properties to ES jvm.options configuration file.	**${ES_JAVA_OPTS}**
${APPLICATION_NAME}-amq	**AMQ_USER**	The username to connect to the JMS broker.	**${APPFORMER_JMS_BROKER_USER}**
	AMQ_PASSWORD	The password to connect to the JMS broker.	**${APPFORMER_JMS_BROKER_PASSWORD}**

Deployment	Variable name	Description	Example value
	AMQ_ROLE	User role for standard broker user.	**${AMQ_ROLE}**
	AMQ_NAME	The name of the broker	**${AMQ_NAME}**
	AMQ_TRANSPORTS	—	openwire
	AMQ_GLOBAL_MAX_SIZE	Maximum amount of memory which message data may consume (Default: Undefined, half of the system's memory).	**${AMQ_GLOBAL_MAX_SIZE}**

4.2.2.3.3.7. 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
${APPLICATION_NAME}-rhdmindex	**${APPLICATION_NAME}-rhdmindex-pvol**	**/opt/elasticsearch/data**	rhdmindex	false

4.2.2.4. External Dependencies

4.2.2.4.1. Volume Claims

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

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

4.2.2.4.2. Secrets

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

decisioncentral-app-secret kieserver-app-secret

4.2.2.4.3. 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 4.1. 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
```

4.3. RHDM72-KIESERVER.YAML TEMPLATE

Application template for a managed KIE Server, for Red Hat Decision Manager 7.2

4.3.1. Parameters

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

Variable name	Image Environment Variable	Description	Example value	Required
APPLICATION_NAME	—	The name for the application.	myapp	True
MAVEN_REPO_ID	**EXTERNAL_MAVEN_REPO_ID**	The id to use for the maven repository, if set. Default is generated randomly.	my-repo-id	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**	Username to access the Maven repository, if required.	—	False
MAVEN_REPO_PASSWORD	**EXTERNAL_MAVEN_REPO_PASSWORD**	Password to access the Maven repository, if required.	—	False
DECISION_CENTRAL_MAVEN_SERVICE	**RHDMCENTR_MAVEN_REPO_SERVICE**	The OpenShift service name for the optional decision central (for maven repo usage), if required	myapp-rhdmcentr	False
DECISION_CENTRAL_MAVEN_USERNAME	**RHDMCENTR_MAVEN_REPO_USERNAME**	Username to access the Maven service hosted by Decision Central inside EAP.	mavenUser	False
DECISION_CENTRAL_MAVEN_PASSWORD	**RHDMCENTR_MAVEN_REPO_PASSWORD**	Password to access the Maven service hosted by Decision Central inside EAP.	maven1!	False
KIE_ADMIN_USER	**KIE_ADMIN_USER**	KIE administrator username	adminUser	False
KIE_ADMIN_PWD	**KIE_ADMIN_PWD**	KIE administrator password	—	False
KIE_SERVER_USER	**KIE_SERVER_USER**	KIE server username (Sets the org.kie.server.user system property)	executionUser	False
KIE_SERVER_PWD	**KIE_SERVER_PWD**	KIE server password (Sets the org.kie.server.pwd system property)	—	False

Variable name	Image Environment Variable	Description	Example value	Required
IMAGE_STREAM_NAMESPACE	—	Namespace in which the ImageStreams for Red Hat Middleware images are installed. These ImageStreams are normally installed in the openshift namespace. You should only need to modify this if you installed the ImageStreams in a different namespace/project.	openshift	True
KIE_SERVER_IMAGE_STREAM_NAME	—	The name of the image stream to use for KIE server. Default is "rhdm72-kieserver-openshift".	rhdm72-kieserver-openshift	True
IMAGE_STREAM_TAG	—	A named pointer to an image in an image stream. Default is "1.1".	1.1	True
KIE_SERVER_CONTROLLER_USER	**KIE_SERVER_CONTROLLER_USER**	KIE server controller username (Sets the org.kie.server.controller.user system property)	controllerUser	False
KIE_SERVER_CONTROLLER_PWD	**KIE_SERVER_CONTROLLER_PWD**	KIE server controller password (Sets the org.kie.server.controller.pwd system property)	—	False

Variable name	Image Environment Variable	Description	Example value	Required
KIE_SERVER_CONTROLLER_TOKEN	**KIE_SERVER_CONTROLLER_TOKEN**	KIE server controller token for bearer authentication (Sets the org.kie.server.controller.token system property)	—	False
KIE_SERVER_CONTROLLER_SERVICE	**KIE_SERVER_CONTROLLER_SERVICE**	The service name for the optional standalone controller. The application uses this service name to register with the controller. (If set, will be used to discover host and port)	—	False
KIE_SERVER_CONTROLLER_HOST	**KIE_SERVER_CONTROLLER_HOST**	KIE server controller host (Used to set the org.kie.server.controller system property)	my-app-controller-ocpuser.os.example.com	False
KIE_SERVER_CONTROLLER_PORT	**KIE_SERVER_CONTROLLER_PORT**	KIE server controller port (Used to set the org.kie.server.controller system property)	8080	False
DROOLS_SERVER_FILTER_CLASSES	**DROOLS_SERVER_FILTER_CLASSES**	KIE server class filtering (Sets the org.drools.server.filter.classes system property)	true	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
KIE_SERVER_HOSTNAME_HTTP	**HOSTNAME_HTTP**	Custom hostname for http service route. Leave blank for default hostname, e.g.: <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.: secure-<application-name>-kieserver-<project>.<default-domain-suffix>	—	False
KIE_SERVER_USE_SECURE_ROUTE_NAME	**KIE_SERVER_USE_SECURE_ROUTE_NAME**	If true, will use secure-APPLICATION_NAME-kieserver vs. APPLICATION_NAME-kieserver as the route name.	false	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**	KIE server bypass auth user (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 in format: containerId=groupId:artifactId:version|c2=g2:a2:v2	rhdm-kieserver-library=org.openshift.quickstarts:rhdm-kieserver-library:1.4.0-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
KIE_SERVER_STARTUP_STRATEGY	**KIE_SERVER_STARTUP_STRATEGY**	When set to LocalContainersStartupStrategy, allows KIE server to start up and function with local config, even when a controller is configured and unavailable.	LocalContainersStartupStrategy	False

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

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

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

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

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

4.3.2. 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.

4.3.2.1. Services

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

Service	Port	Name	Description
${APPLICATION_NAME}-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.

4.3.2.2. 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. Refer to the Openshift documentation for more information.

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

4.3.2.3. 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. Refer to the Openshift documentation for more information.

4.3.2.3.1. Triggers

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

Deployment	Triggers
${APPLICATION_NAME}-kieserver	ImageChange

4.3.2.3.2. 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. Refer to the container-engine documentation for more information.

Deployment	Replicas
${APPLICATION_NAME}-kieserver	1

4.3.2.3.3. Pod Template

4.3.2.3.3.1. Service Accounts

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

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

4.3.2.3.3.2. Image

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

4.3.2.3.3.3. Readiness Probe

${APPLICATION_NAME}-kieserver

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/services/rest/server/readycheck
```

4.3.2.3.3.4. Liveness Probe

${APPLICATION_NAME}-kieserver

```
/bin/bash -c curl --fail --silent -u '${KIE_ADMIN_USER}:${KIE_ADMIN_PWD}'
http://localhost:8080/services/rest/server/readycheck
```

4.3.2.3.3.5. Exposed Ports

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

4.3.2.3.3.6. Image Environment Variables

Deployment	Variable name	Description	Example value
${APPLICATION_NAME}-kieserver	**DROOLS_SERVER_FILTER_CLASSES**	KIE server class filtering (Sets the org.drools.server.filter.classes system property)	**${DROOLS_SERVER_FILTER_CLASSES}**
	KIE_ADMIN_USER	KIE administrator username	**${KIE_ADMIN_USER}**
	KIE_ADMIN_PWD	KIE administrator password	**${KIE_ADMIN_PWD}**
	KIE_MBEANS	KIE server mbeans enabled/disabled (Sets the kie.mbeans and kie.scanner.mbeans system properties)	**${KIE_MBEANS}**
	KIE_SERVER_BYPASS_AUTH_USER	KIE server bypass auth user (Sets the org.kie.server.bypass.auth.user system property)	**${KIE_SERVER_BYPASS_AUTH_USER}**
	KIE_SERVER_CONTROLLER_USER	KIE server controller username (Sets the org.kie.server.controller.user system property)	**${KIE_SERVER_CONTROLLER_USER}**
	KIE_SERVER_CONTROLLER_PWD	KIE server controller password (Sets the org.kie.server.controller.pwd system property)	**${KIE_SERVER_CONTROLLER_PWD}**

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)	**${KIE_SERVER_CONTROLLER_TOKEN}**
	KIE_SERVER_CONTROLLER_SERVICE	The service name for the optional standalone controller. The application uses this service name to register with the controller. (If set, will be used to discover host and port)	**${KIE_SERVER_CONTROLLER_SERVICE}**
	KIE_SERVER_CONTROLLER_PROTOCOL	—	ws
	KIE_SERVER_CONTROLLER_HOST	KIE server controller host (Used to set the org.kie.server.controller system property)	**${KIE_SERVER_CONTROLLER_HOST}**
	KIE_SERVER_CONTROLLER_PORT	KIE server controller port (Used to set the org.kie.server.controller system property)	**${KIE_SERVER_CONTROLLER_PORT}**
	KIE_SERVER_ID	—	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_ROUTE_NAME	—	**${APPLICATION_NAME}-kieserver**
	KIE_SERVER_USE_SECURE_ROUTE_NAME	If true, will use secure-APPLICATION_NAME-kieserver vs. APPLICATION_NAME-kieserver as the route name.	**${KIE_SERVER_USE_SECURE_ROUTE_NAME}**
	KIE_SERVER_USER	KIE server username (Sets the org.kie.server.user system property)	**${KIE_SERVER_USER}**

Deployment	Variable name	Description	Example value
	KIE_SERVER_PWD	KIE server password (Sets the org.kie.server.pwd system property)	**${KIE_SERVER_PWD}**
	KIE_SERVER_CONTAINER_DEPLOYMENT	KIE Server Container deployment configuration in format: containerId=groupId:artifactId:version	c2=g2:a2:v2
	${KIE_SERVER_CONTAINER_DEPLOYMENT}	**MAVEN_REPOS**	—
	RHDMCENTR,EXTERNAL	**RHDMCENTR_MAVEN_REPO_SERVICE**	The OpenShift service name for the optional decision central (for maven repo usage), if required
	${DECISION_CENTRAL_MAVEN_SERVICE}	**RHDMCENTR_MAVEN_REPO_PATH**	—
	/maven2/	**RHDMCENTR_MAVEN_REPO_USERNAME**	Username to access the Maven service hosted by Decision Central inside EAP.
	${DECISION_CENTRAL_MAVEN_USERNAME}	**RHDMCENTR_MAVEN_REPO_PASSWORD**	Password to access the Maven service hosted by Decision Central inside EAP.
	${DECISION_CENTRAL_MAVEN_PASSWORD}	**EXTERNAL_MAVEN_REPO_ID**	The id to use for the maven repository, if set. Default is generated randomly.
	${MAVEN_REPO_ID}	**EXTERNAL_MAVEN_REPO_URL**	Fully qualified URL to a Maven repository or service.
	${MAVEN_REPO_URL}	**EXTERNAL_MAVEN_REPO_USERNAME**	Username to access the Maven repository, if required.

Deployment	Variable name	Description	Example value
	${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**	When set to LocalContainersStartupStrategy, allows KIE server to start up and function with local config, even when a controller is configured and unavailable.
	${KIE_SERVER_STARTUP_STRATEGY}	**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}	**JGROUPS_PING_PROTOCOL**	—
	openshift.DNS_PING	**OPENSHIFT_DNS_PING_SERVICE_NAME**	—
	${APPLICATION_NAME}-kieserver-ping	**OPENSHIFT_DNS_PING_SERVICE_PORT**	—

Deployment	Variable name	Description	Example value
	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 Username used to create the Client if it doesn't exist
	${SSO_USERNAME}	**SSO_PASSWORD**	RH-SSO Realm Admin Password used to create the Client
	${SSO_PASSWORD}	**SSO_DISABLE_SSL_CERTIFICATE_VALIDATION**	RH-SSO Disable SSL Certificate Validation
	${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}	**SSO_PRINCIPAL_ATTRIBUTE**	RH-SSO Principal Attribute to use as username.
	${SSO_PRINCIPAL_ATTRIBUTE}	**HOSTNAME_HTTP**	Custom hostname for http service route. Leave blank for default hostname, e.g.: <application-name>-kieserver-<project>.<default-domain-suffix>

Deployment	Variable name	Description	Example value
	${KIE_SERVER_HOSTNAME_HTTP}	**HOSTNAME_HTTPS**	Custom hostname for https service route. Leave blank for default hostname, e.g.: secure-<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}	**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.

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

Deployment	Variable name	Description	Example value
	${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}	**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

Deployment	Variable name	Description	Example value
	${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}	**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.

Deployment	Variable name	Description	Example value
	${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}	**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.

4.3.2.3.3.7. Volumes

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

4.3.2.4. External Dependencies

4.3.2.4.1. Secrets

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

kieserver-app-secret

4.4. 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 run 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 run 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 run 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 Wednesday, February 13, 2019.