



Red Hat Decision Manager 7.3

Operator を使用した Red Hat OpenShift Container Platform への Red Hat Decision Manager 環境のデプロイメント

ガイド

Red Hat Decision Manager 7.3 Operator を使用した Red Hat OpenShift Container Platform への Red Hat Decision Manager 環境のデプロイメント

ガイド

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Deploying_a_Red_Hat_Decision_Manager_environment_on_Red_Hat_OpenShift_Container_Platform file |.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

概要

本書は、Red Hat OpenShift Container Platform に Operator を使用して Red Hat Decision Manager 7.3 環境をデプロイする方法を説明します。

目次

はじめに	3
第1章 RED HAT OPENSIFT CONTAINER PLATFORM における RED HAT DECISION MANAGER の概要	4
第2章 OPENSIFT 環境に RED HAT DECISION MANAGER をデプロイする準備	5
2.1. RED HAT レジストリーに対してお使いの環境が認証されていることを確認する方法	5
2.2. DECISION SERVER にシークレットの作成	5
2.3. BUSINESS CENTRAL へのシークレットの作成	6
第3章 OPENSIFT OPERATOR を使用した RED HAT DECISION MANAGER 環境のデプロイと管理	7
3.1. BUSINESS AUTOMATION OPERATOR のサブスクリプション	7
3.2. OPERATOR を使用した RED HAT DECISION MANAGER 環境のデプロイ	7
3.3. OPERATOR を使用してデプロイした環境の変更	11
3.4. LDAP ロールマッピングファイルの指定	12
付録A バージョン情報	14

はじめに

システムエンジニアは、Red Hat OpenShift Container Platform に Red Hat Decision Manager 環境をデプロイしてプロセスや他のビジネスアセットを開発または実行するインフラストラクチャーを提供します。OpenShift Operator を使用して、構造化された YAML ファイルに定義された環境をデプロイして、必要に応じてこの環境を維持して変更できます。



注記

この機能は現在、テクノロジープレビュー機能のみとなっています。Red Hat のテクノロジープレビュー機能のサポートの詳細は、[テクノロジープレビュー機能のサポート範囲](#)を参照してください。

前提条件

- OpenShift 環境で 4 ギガバイト以上のメモリーが利用できる。
- OpenShift Operator Framework が OpenShift 環境にインストールされており、起動している。
- デプロイメントする OpenShift プロジェクトが作成されている。
- OpenShift Web コンソールを使用してプロジェクトにログインしている。
- 動的永続ボリューム (PV) のプロビジョニングが有効になっている。または、動的 PV プロビジョニングが有効でない場合は、十分な永続ボリュームが利用できる状態でなければなりません。デフォルトでは、以下のサイズが必要です。
 - デフォルトでは、Business Central は 1 Gi 分の PV が必要です。Business Central 永続ストレージの PV サイズを変更できます。
- Business Central の Pod をスケーリングする予定がある場合、OpenShift 環境では、**ReadWriteMany** モードで永続ボリュームがサポートされている。



重要

ReadWriteMany モードは、OpenShift Online および OpenShift Dedicated ではサポートされません。

第1章 RED HAT OPENSIFT CONTAINER PLATFORM における RED HAT DECISION MANAGER の概要

Red Hat Decision Manager は、Red Hat OpenShift Container Platform 環境にデプロイすることができます。

この場合、Red Hat Decision Manager のコンポーネントは、別の OpenShift Pod としてデプロイされます。各 Pod のスケールアップとダウンを個別に行い、特定のコンポーネントに必要な数だけコンテナを提供できます。標準の OpenShift の手法を使用して Pod を管理し、負荷を分散できます。

以下の Red Hat Decision Manager の主要コンポーネントが OpenShift で利用できます。

- Decision Server (**実行サーバー (Execution Server)** または **KIE Server** と呼ばれる) は、インフラストラクチャーの要素でデシジョンサービスやその他のデプロイ可能なアセットを実行します (これらすべてで総称で **サービス** と呼ぶ)。サービスのすべてのロジックは実行サーバーで実行されます。

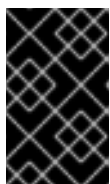
Decision Server Pod は自由にスケールアップして、同一または異なるホストで実行するコピーに必要な数だけ提供できます。Pod をスケールアップまたはスケールダウンすると、そのコピーはすべて同じプロセスで実行します。OpenShift は負荷分散を提供しているため、要求はどの Pod でも処理できます。

個別の Decision Server Pod をデプロイして、異なるサービスグループを実行することができます。この Pod もスケールアップやスケールダウンが可能です。個別の複製 Decision Server Pod を必要な数だけ設定することができます。

- Business Central は、オーサリングサービス用の Web ベースの対話環境で、Business Central では管理コンソールが提供されています。Business Central は管理コンソールも提供します。Business Central を使用してサービスを開発し、それらを Decision Server にデプロイできます。

Business Central は一元化アプリケーションです。複数の Pod を実行し、同じデータを共有する高可用性用に設定できます。

Business Central には開発するサービスのソースを保管する Git リポジトリが含まれます。また、ビルトインの Maven リポジトリも含まれます。設定に応じて、Business Central はコンパイルしたサービス (KJAR ファイル) をビルドイン Maven リポジトリに配置できます (設定した場合は外部 Maven リポジトリにも可能)。



重要

現在のバージョンでは、高可用性の Business Central 機能はテクノロジープレビュー機能となっています。Red Hat のテクノロジープレビュー機能のサポートの詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

OpenShift 内でさまざまな環境設定にこのコンポーネントおよびその他のコンポーネントを配置できます。

第2章 OPENSIFT 環境に RED HAT DECISION MANAGER をデプロイする準備

OpenShift 環境に Red Hat Decision Manager をデプロイする前に、準備タスクをいくつか完了する必要があります。追加イメージ (たとえば、デシジョンサービスの新しいバージョン、または別のデシジョンサービス) をデプロイする場合は、このタスクを繰り返す必要はありません。

2.1. RED HAT レジストリーに対してお使いの環境が認証されていることを確認する方法

Red Hat OpenShift Container Platform で Red Hat Decision Manager コンポーネントをデプロイするには、OpenShift が Red Hat レジストリーから正しいイメージをダウンロードできるようにする必要があります。OpenShift は、お使いのサービスアカウントのユーザー名とパスワードを使用して Red Hat レジストリーへの認証が行われるように設定する必要があります。

手順

1. Red Hat OpenShift Container Platform が Red Hat レジストリーへのアクセス用に、ユーザー名とパスワードで設定されているかを判断します。必須の設定に関する詳細は、[レジストリーの場所の設定](#) を参照してください。OpenShift オンラインサブスクリプションを使用する場合は、Red Hat レジストリー用のアクセスはすでに設定されています。
2. Red Hat OpenShift Container Platform は、ユーザー名とパスワードで Red Hat レジストリーにアクセスするように設定した場合には、これ以上何もする必要はありません。設定していない場合には、以下の手順を実行してください。
 - a. **oc** コマンドで OpenShift にログインして、プロジェクトがアクティブであることを確認します。
 - b. [Registry Service Accounts for Shared Environments](#) で説明されている手順を実行します。Red Hat カスタマーポータルにログインして、このドキュメントにアクセスし、レジストリーサービスアカウントを作成する手順を実行します。
 - c. **OpenShift Secret** タブを選択し、**Download secret** のリンクをクリックして、YAML シークレットファイルをダウンロードします。
 - d. ダウンロードしたファイルを確認して、**name:** エントリーに記載の名前をメモします。
 - e. 以下のコマンドを実行します。

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

<file_name> はダウンロードしたファイルに、<secret_name> はファイルの **name:** のエントリーに記載されている名前に置き換えてください。

2.2. DECISION SERVER にシークレットの作成

OpenShift は、シークレットと呼ばれるオブジェクトを使用してパスワードやキーストアなどの機密情報を保持します。OpenShift のシークレットに関する詳細は、OpenShift ドキュメントの [シークレット](#) の章を参照してください。

Decision Server では HTTPS アクセスに SSL 証明書を使用します。このデプロイメントでは、サンプルシークレットを自動的に作成できます。ただし、実稼働環境では、Decision Server の SSL 証明書を作成し、これをシークレットとして OpenShift 環境に提供する必要があります。

手順

1. Decision Server の SSL 暗号化の秘密鍵および公開鍵を使用して SSL キーストアを生成します。プロダクション環境で、期待する Decision Server の URL に一致する有効な署名付き証明書を生成します。キーストアを **keystore.jks** ファイルに保存します。証明書の名前と、キーストアファイルのパスワードを記録します。
自己署名または購入した SSL 証明書でキーストアを作成する方法は、[SSL 暗号化キーおよび証明書](#) を参照してください。
2. **oc** コマンドを使用して、新しいキーストアファイルからシークレット **kieserver-app-secret** を生成します。

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

2.3. BUSINESS CENTRAL へのシークレットの作成

OpenShift 環境に Business Central をデプロイする場合は、このコンポーネントは SSL 証明書を使用して HTTPS アクセスができるようにする点に注意してください。このデプロイメントでは、サンプルシークレットを自動的に作成できます。ただし、実稼働環境では、Business Central の SSL 証明書を作成し、これをシークレットとして OpenShift 環境に提供する必要があります。Business Central と Decision Server に同じ証明書およびキーストアを使用しないでください。

手順

1. Business Central の SSL 暗号化の秘密鍵および公開鍵を使用して、SSL キーストアを生成します。実稼働環境で、期待する Business Central の URL に一致する有効な署名付き証明書を生成します。キーストアを **keystore.jks** ファイルに保存します。証明書の名前と、キーストアファイルのパスワードを記録します。
自己署名または購入した SSL 証明書でキーストアを作成する方法は、[SSL 暗号化キーおよび証明書](#) を参照してください。
2. **oc** コマンドを使用して、新しいキーストアファイルからシークレット **decisioncentral-app-secret** を生成します。

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

第3章 OPENSIFT OPERATOR を使用した RED HAT DECISION MANAGER 環境のデプロイと管理

OpenShift Operator を使用して Red Hat Decision Manager 環境をデプロイするには、YAML ソースとして環境の設定を行う必要があります。

Red Hat OpenShift Container Platform で環境をデプロイする場合には、環境の YAML 記述を作成し、環境が常にこの記述と一致していることを確認します。記述を編集して環境を変更することができます。

3.1. BUSINESS AUTOMATION OPERATOR のサブスクリプション

Operator を使用して Red Hat Decision Manager をデプロイできるようにするには、OpenShift のビジネス自動化のオペレーターにサブスクリプション登録する必要があります。Operator がカタログで利用できない場合は、これをダウンロードし、インストールする必要があります。

手順

1. OpenShift Web クラスターコンソールでプロジェクトに移動します。
2. OpenShift Web コンソールのナビゲーションパネルで、**Operators** を選択してから **Catalog Sources** を選択します。
3. **ビジネスオートメーション** を検索します。見つかった場合は、その横にある **Create Subscription** をクリックします。
4. **Business Automation** エントリーがない場合には、以下の手順を実行してください。
 - a. 管理者として OpenShift 環境にログインします。
 - b. 以下のコマンドを入力します。

```
$ oc create -f https://raw.githubusercontent.com/kiogroup/kie-cloud-operator/1.0.1/deploy/catalog\_resources/redhat/catalog-source.yaml
```
 - c. ブラウザーで OpenShift Web コンソールを更新します。
 - d. **Business Automation** を再度検索します。その横にある **Create Subscription** をクリックします。
5. 新規サブスクリプションの YAML 記述が表示されます。**Create** をクリックして、サブスクリプションを作成してください。

3.2. OPERATOR を使用した RED HAT DECISION MANAGER 環境のデプロイ

オペレーターにサブスクリプション登録した後に、**アプリケーションを作成して** 環境をデプロイできます。

手順

1. OpenShift Web クラスターコンソールでプロジェクトに移動します。

2. OpenShift Web コンソールのナビゲーションパネルで、**Operators** を選択してから **Subscriptions** を選択します。
3. **businessautomation** を含むサブスクリプションの名前をクリックします。このサブスクリプションに関する情報が表示されます。
4. **Installed version** 見出しの下で、サブスクリプションのバージョン名をクリックします。サブスクリプションの概要が表示されます。
5. **Create KieApp** をクリックします。YAML ソースが表示されます。
6. **name** フィールドに、プロジェクトで一意的なアプリケーション名を設定します。
7. **environment** フィールドには、必須の環境タイプを設定します。これらのタイプはそれぞれ、デフォルトのデプロイメントパターンです。このパターンは、YAML ソースを編集して変更できます。完了後でもデプロイメントを変更できます。利用可能なタイプは以下のとおりです。
 - **rhdm-trial**: すばやく設定して、アセットの開発や実行を評価またはデモで確認するのに使用できる試用版の環境。Business Central と Decision Server 1 台が含まれています。この環境では永続ストレージを使用しないため、この環境で実行した作業内容は保存されません。
 - **rhdm-authoring**: Business Central を使用してサービスを作成し、変更する環境。この環境は、オーサリングの作業用の Business Central と、サービスのテスト実行用の Decision Server 1 台で設定されます。この環境を使用して、ステージングおよび実稼働の目的でサービスを実行することも可能です。環境に Decision Server を追加して、同じ Business Central で管理できます。
 - **rhdm-authoring-ha**: Business Central を使用してサービスを作成し、変更する環境。この環境は、オーサリングの作業用の Business Central と、サービスのテスト実行用の Decision Server 1 台で設定されます。このバージョンのオーサリング環境は、高可用性が確保されるように Business Central Pod のスケーリングをサポートします。
 - **rhdm-production-immutable**: ステージングおよび実稼働目的で既存のサービスを実行するための別の環境。Decision Server の複製 Pod を 1 つまたは複数設定して、ソースからサービスを構築できます。この環境では、Process Server の Pod のデプロイ時に、サービスまたはサービスグループをロードおよび起動するイメージをビルドします。この Pod でサービスを停止したり、新しいサービスを追加したりすることはできません。サービスの別のバージョンを使用したり、別の方法で設定を変更する必要がある場合は、新規のサーバーイメージをデプロイして、古いサーバーと入れ替えます。このシステムでは、Decision Server は OpenShift 環境の他の Pod のように実行されます。コンテナベースの統合ワークフローを使用でき、他のツールを使用して Pod を管理する必要はありません。
8. 本書に参考用として記載されているスニペットを使用し、YAML ソースに行を追加して、環境を変更します。
 - 以下のスニペットは、イミュータブルの Decision Server Pod を追加して、ソースからサービスを構築します。イミュータブルな環境を作成する場合には、このスニペットのコピーを 1 つ以上追加する必要があります。

```
objects:
  servers:
    - build:
        kieServerContainerDeployment: <deployment>
        gitSource:
```

```
uri: <url>
reference: <branch>
contextDir: <directory>
```

以下の値を置き換えます。

- **<deployment>**: ソースからビルドしたデシジョンサービス (KJAR ファイル) の識別情報。形式は **<containerId>=<groupId>:<artifactId>:<version>** になります。区切り記号 | を使用して 2 つ以上の KJAR ファイルを指定できます (例: **containerId=groupId:artifactId:version|c2=g2:a2:v2**)。Maven ビルドプロセスは、Git リポジトリのソースからこのようなファイルをすべて生成する必要があります。
- **<url>**: デシジョンサービスのソースを含む Git リポジトリの URL。
- **<branch>**: Git リポジトリのブランチ。
- **<directory>**: Git リポジトリからダウンロードしたプロジェクトのソースへのパス。
- 以下のスニペットでは、Decision Server の数と設定を設定し、お使いの環境にすでに存在する Business Central で管理します。このスニペットでは、3 種の名前セットの中にサーバー 6 台が含まれています。

```
apiVersion: app.kiegroup.org/v1
kind: KieApp
metadata:
  name: server-config
spec:
  environment: <environment_type>
  objects:
    console:
      env:
        - name: MY_VALUE
          value: "example"
    servers:
      # Kieserver sets will be named sequentially server-config-kieserver1, server-config-
      kieserver1-2
      - deployments: 2
        # Env variables that will be added to all the kie servers in this set
        env:
          - name: MY_VALUE
            value: "example"
        # Override default memory limits for all the kie servers in this set
        resources:
          limits:
            memory: 2Gi
      # Kieserver sets will be named sequentially server-config-kieserver2, server-config-
      kieserver2-2
      - deployments: 2
        # Env variables that will be added to all the kie servers in this set
        env:
          - name: MY_VALUE
            value: "example"
      # Kieserver sets will be named sequentially server, server-2
      - name: server
        deployments: 2
        env:
```

```

- name: MY_VALUE
  value: "example"
# Override default memory limits for all the kie servers in this set
resources:
  limits:
    memory: 2Gi

```

<environment_type> は、設定する環境タイプに置き換えます。

- 以下のスニペットでは、実稼働環境での要件に応じて、HTTPS 通信の既存のシークレットを使用して Decision Servers と Business Central を設定します。この例では、**server-a-keystore** シークレットを使用して2つのサーバーが作成されます。(シークレットの作成手順については、「[Business Central へのシークレットの作成](#)」と「[Decision Server にシークレットの作成](#)」を参照してください。)

```

apiVersion: app.kiegroup.org/v1
kind: KieApp
metadata:
  name: keystore-config
spec:
  environment: <environment_type>
  objects:
    console:
      keystoreSecret: console-keystore
    servers:
      - name: server-a
        deployments: 2
        keystoreSecret: server-a-keystore
      - name: server-b
        keystoreSecret: server-b-keystore

```

<environment_type> は、設定する環境タイプに置き換えます。

- 以下のスニペットでは、管理者ユーザー (**admin**) のパスワードと、アプリケーション名 (**app2**) を設定します。

```

commonConfig:
  adminPassword: password
  applicationName: app2

```

- 以下のスニペットは LDAP 認証を設定します。このパラメーターは Red Hat JBoss EAP の LdapExtended ログインモジュールに対応します。これらの設定に関する説明は、[LdapExtended ログインモジュール](#) を参照してください。

```

auth:
  ldap:
    url: ldaps://myldap.example.com
    bindDN: uid=admin,ou=users,ou=example,ou=com
    bindCredential: s3cret
    baseCtxDN: ou=users,ou=example,ou=com
    baseFilter: (uid={0})
    searchScope: SUBTREE_SCOPE
    roleAttributeID: memberOf
    rolesCtxDN: ou=groups,ou=example,ou=com
    roleFilter: (memberOf={1})

```

```

defaultRole: guest
roleMapper:
  rolesProperties: /conf/roleMapper.properties
  replaceRole: true

```

LDAP サーバーがデプロイメントに必要な全ロールを定義していない場合は、LDAP グループを Red Hat Decision Manager ロールにマッピングしてください。LDAP ロールマッピングを有効にするに

は、`/opt/eap/standalone/configuration/rolemapping/rolemapping.properties` など、ロールマッピングを定義するファイルの完全修飾パス名に、`rolesProperties` の値を設定してください。このファイルを指定して、適用可能なデプロイメント設定のこのパスにマウントする必要があります。このファイルを指定する方法は、「[LDAP ロールマッピングファイルの指定](#)」を参照してください。

`replaceRole` パラメーターが `true` に設定されている場合には、LDAP サーバーに定義されているロールは、マッピングされたロールに置き換えられます。このパラメーターが `false` に設定されている場合には、マッピングされたロールおよび LDAP サーバーに定義されているロール両方がユーザーアプリケーションロールとして設定されます。デフォルトの設定は `false` です。

9. YAML ソースの変更を完了したら、**Create** をクリックしてアプリケーションを作成します。



注記

たとえば、<https://github.com/kiegroup/kie-cloud-operator/tree/1.0.1/deploy/examples> で異なるデータベースサーバーを使用するサーバー用の他の設定サンプルを表示できます。

3.3. OPERATOR を使用してデプロイした環境の変更

環境を Operator を使用してデプロイした場合は、通常の OpenShift の手法を使用して環境を変更することはできません。たとえば、Pod を削除するには、同じパラメーターで自動的に再作成されます。

環境を変更するには、環境の YAML の記述を変更する必要があります。パスワードなどの一般的な設定を変更し、Decision Server を追加してスケーリングできます。

手順

1. OpenShift Web クラスターコンソールでプロジェクトに移動します。
2. OpenShift Web コンソールのナビゲーションパネルで、**Operators** を選択してから **Subscriptions** を選択します。
3. **businessautomation** を含むサブスクリプションの名前をクリックします。このサブスクリプションに関する情報が表示されます。
4. **Installed version** 見出しの下で、サブスクリプションのバージョン名をクリックします。サブスクリプションの概要が表示されます。
5. **Instances** タブを選択します。
6. デプロイした環境の名前をクリックします。
7. **YAML** タブを選択します。YAML ソースが表示されます。
8. パスワードなどの共通の設定を変更するには、`commonConfig:` の値を編集します。

9. 新しい Decision Server を追加する場合は、以下の例に示されているように、**servers:** のブロックの最後に対応の記述を追加します。

- 名前が **server-a** と **server-a-2** のサーバー 2 台を追加するには、以下の行を追加します。

```
- deployments: 2
  name: server-a
```

- S2I プロセスのソースから構築するサービスを含む、イミュータブルな Decision Server を追加するには、以下の行を追加します。

```
- build:
  kieServerContainerDeployment: <deployment>
  gitSource:
    uri: <url>
    reference: <branch>
    contextDir: <directory>
```

以下の値を置き換えます。

- **<deployment>**: ソースからビルドしたデシジョンサービス (KJAR ファイル) の識別情報。形式は **<containerId>=<groupId>:<artifactId>:<version>** になります。区切り記号 | を使用して 2 つ以上の KJAR ファイルを指定できます (例: **containerId=groupId:artifactId:version|c2=g2:a2:v2**)。Maven ビルドプロセスは、Git リポジトリのソースからこのようなファイルをすべて生成する必要があります。
 - **<url>**: デシジョンサービスのソースを含む Git リポジトリの URL。
 - **<branch>**: Git リポジトリのブランチ。
 - **<directory>**: Git リポジトリからダウンロードしたプロジェクトのソースへのパス。
10. Decision Server をスケーリングする場合は、**servers:** のブロックに含まれるサーバーの記述を検索して、その記述の下に **replicas:** 設定を追加します。たとえば、**replicas: 3** はサーバーを Pod 3 つにスケーリングします。
11. **Save Changes** をクリックします。
12. **This object has been updated** のポップアップメッセージが表示されるまで待機します。
13. **Reload** をクリックして、環境の新しい YAML の記述を表示します。

3.4. LDAP ロールマッピングファイルの指定

AUTH_ROLE_MAPPER_ROLES_PROPERTIES パラメーターを設定する場合は、ロールマッピングを定義するファイルを指定する必要があります。影響を受けるすべてのデプロイメント設定にこのファイルをマウントしてください。

手順

1. **my-role-map** など、ロールマッピングのプロパティファイルを作成します。ファイルには、次の形式のエントリが含まれている必要があります。

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


以下に例を示します。

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

2. 以下のコマンドを入力して、このファイルから OpenShift 設定ファイルのマッピングを作成します。

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

<new_name> は、Pod に指定するファイルの名前 (**AUTH_ROLE_MAPPER_ROLES_PROPERTIES** ファイルで指定した名前と同じである必要があります) に置き換えます。また、**<existing_name>** は、作成したファイル名に置き換えます。以下に例を示します。

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

3. ロールマッピング用に指定した全デプロイメント設定に設定マップをマウントします。すべてのデプロイメント設定について、以下のコマンドを実行します。

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

<mapping_dir> は、**/opt/eap/standalone/configuration/rolemapping** など、**AUTH_ROLE_MAPPER_ROLES_PROPERTIES** で設定したディレクトリー名 (ファイル名なし) に置き換えます。

付録A バージョン情報

本書の最終更新日: 2021年11月15日(月)