



Red Hat Process Automation Manager 7.4

Red Hat OpenShift Container Platform への
Red Hat Process Automation Manager フリー
フォーム管理サーバー環境のデプロイ

ガイド

Red Hat Process Automation Manager 7.4 Red Hat OpenShift Container Platform への Red Hat Process Automation Manager フリーフォーム管理サーバー環境のデプロイ

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_Process_Automation_Manager_freeform_managed_server_environment 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 Red Hat Process Automation Manager 7.4 フリーフォーム管理サーバー環境をデプロイする方法を説明します。

目次

前書き	4
第1章 RED HAT OPENSIFT CONTAINER PLATFORM における RED HAT PROCESS AUTOMATION MANAGER の概要	5
第2章 OPENSIFT 環境に RED HAT PROCESS AUTOMATION MANAGER をデプロイする準備	8
2.1. イメージストリームとイメージレジストリーの可用性確認	8
2.2. PROCESS SERVER にシークレットの作成	9
2.3. BUSINESS CENTRAL へのシークレットの作成	10
2.4. GLUSTERFS 設定の変更	10
2.5. 外部データベースのカスタム PROCESS SERVER 拡張イメージのビルド	12
2.6. オフラインで使用する MAVEN ミラーリポジトリの用意	14
第3章 フリーフォーム管理サーバー環境	16
3.1. フリーフォーム環境用の MONITORING および単一 PROCESS SERVER のデプロイ	16
3.1.1. Monitoring および単一の Process Server 用のテンプレート設定の開始	16
3.1.2. Monitoring および単一の Process Server に必要なパラメーターの設定	17
3.1.3. Monitoring および単一の Process Server の Pod レプリカ数の設定	18
3.1.4. 公開インターネットへの接続のない環境で Monitoring および単一の Process Server の Maven ミラーへのアクセス設定	19
3.1.5. Monitoring および単一の Process Server の RH-SSO 認証用パラメーター設定	20
3.1.6. Monitoring および単一の Process Server の LDAP 認証用パラメーター設定	22
3.1.7. Monitoring および単一の Process Server の Prometheus メトリクス収集の有効化	23
3.1.8. Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する	23
3.2. フリーフォーム環境用の追加の管理 PROCESS SERVER のデプロイ	24
3.2.1. 追加の管理 Process Server のテンプレート設定の開始	24
3.2.2. 追加の管理 Process Server に必要なパラメーターの設定	25
3.2.3. 追加の管理 Process Server のイメージストリーム namespace の設定	26
3.2.4. 追加の管理 Process Server 用の Business Central Monitoring インスタンスについての情報の設定	27
3.2.5. 追加の管理 Process Server の公開インターネットへの接続のない環境に Maven ミラーへのアクセスする手順	27
3.2.6. 追加の管理 Process Server の RH-SSO 認証パラメーターの設定	28
3.2.7. 追加の管理 Process Server の LDAP 認証パラメーターの設定	30
3.2.8. 追加の管理 Process Server 用に外部データベースサーバーを使用するためのパラメーターの設定	31
3.2.9. 追加の管理 Process Server の Prometheus メトリクス収集の有効化	33
3.2.10. 追加の管理 Process Server のテンプレートデプロイメントの開始	33
3.3. (任意) LDAP ロールマッピングファイルの指定	33
第4章 RED HAT PROCESS AUTOMATION MANAGER ロールおよびユーザー	35
第5章 OPENSIFT テンプレートの参考資料	37
5.1. RHPAM74-MANAGED.YAML テンプレート	37
5.1.1. パラメーター	37
5.1.2. オブジェクト	54
5.1.2.1. サービス	54
5.1.2.2. ルート	55
5.1.2.3. デプロイメント設定	55
5.1.2.3.1. トリガー	55
5.1.2.3.2. レプリカ	56
5.1.2.3.3. Pod テンプレート	56
5.1.2.4. 外部の依存関係	78
5.1.2.4.1. ボリューム要求	78
5.1.2.4.2. シークレット	78

5.2. OPENSIFT の使用に関するクイックリファレンス	78
付録A バージョン情報	80

前書き

システムエンジニアは、Red Hat OpenShift Container Platform に Red Hat Process Automation Manager フリーフォーム管理サーバー環境をデプロイして、サービス、プロセスアプリケーション、およびその他のビジネスアセットを実行するためのインフラストラクチャーを提供できます。管理 Process Server はいくつでもデプロイでき、この Process Server は同じ Business Central Monitoring を使用して制御できます。必要に応じて Process Server を追加および削除できます。Business Central Monitoring は、Process Server への接続を自動的に確立します。同じ Business Central Monitoring インスタンスを使用して、イミュータブル Process Server を監視することもできます。

前提条件

- OpenShift クラスター/namespace で 4 ギガバイト以上のメモリーが利用可能である。
- デプロイメントに使用する OpenShift プロジェクトが作成されている。
- **oc** コマンドを使用してプロジェクトにログインしている。**oc** コマンドラインツールの詳細は、『[OpenShift Container Platform CLI リファレンス](#)』を参照してください。OpenShift Web コンソールを使用してテンプレートをデプロイするには、Web コンソールを使用してログインしている必要もあります。
- 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。
 - Business Central Monitoring には、64Mi PV が必要です。
 - それぞれの Process Server のデプロイメントには、デフォルトでデータベースに 1Gi PV が 1 つ必要です。テンプレートパラメーターの PV サイズを変更できます。複数の Process Server をデプロイする場合には、それぞれに別個のデータベース PV が必要です。この要件は、外部データベースサーバーを使用する場合には適用されません。
- お使いの OpenShift 環境で **ReadWriteMany** モードを使用した永続ボリュームをサポートしている。OpenShift Online ボリュームプラグインでのアクセスモードのサポートに関する情報は、「[アクセスモード](#)」を参照してください。

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

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

この場合に、Red Hat Process Automation 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 のスケールアップおよびスケール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 Process Automation Manager の主要コンポーネントが OpenShift で利用できます。

- **Process Server (実行サーバー (Execution Server) または KIE Server と呼ばれる)** は、意思決定サービス、プロセスアプリケーションおよびその他のデプロイ可能なアセット (サービスと総称される) を実行するインフラストラクチャー要素です。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。

通常、Process Server にはデータベースサーバーが必要です。別の OpenShift Pod にデータベースサーバーを提供したり、別のデータベースサーバーを使用するように OpenShift で実行サーバーを設定したりできます。また、Process Server では H2 データベースを使用できますが、使用する場合は、Pod をスケールアップできません。

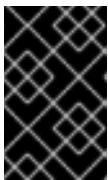
Process Server Pod をスケールアップして、同一または異なるホストで実行するコピーを必要な数だけ提供できます。Pod をスケールアップまたはスケールダウンすると、そのコピーはすべて同じデータベースサーバーを使用し、同じサービスを実行します。OpenShift は負荷分散を提供しているため、要求はどの Pod でも処理できます。

Process Server Pod を個別にデプロイし、サービスの異なる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。複製された個別の Process 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に対する Web ベースのインタラクティブ環境です。また、管理および監視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、Process Server にデプロイできます。また、Business Central を使用してプロセスの実行を監視することもできます。

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。

- **Business Central Monitoring** は Web ベースの管理および監視コンソールです。Process Server へのサービスのデプロイメントを管理し、監視情報を提供しますが、オーサリング機能は含まれません。このコンポーネントを使用して、ステージング環境および実稼働環境を管理できます。
- **Smart Router** は、Process Server と、Process Server と対話するその他のコンポーネントとの

間の任意のレイヤーです。環境に、複数の Process Server で実行するサービスが多数含まれる場合、Smart Router はすべてのクライアントアプリケーションに対応するエンドポイントを1つ提供します。クライアントアプリケーションは、サービスを要求する REST API 呼び出しを実行できます。Smart Router は、特定の要求を処理できる Process Server を自動的に呼び出します。

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。

以下の環境タイプが一般的です。

- **オーサリング:** Business Central を使用してサービスを作成し、変更するために使用する環境です。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 Process Server 1 台で設定されます。この環境のデプロイメント手順については、[『Red Hat OpenShift Container Platform への Red Hat Process Automation Manager オーサリング環境のデプロイ』](#)を参照してください。
- **管理対象のデプロイメント:** ステージングおよび実稼働用として既存のサービスを実行するのに使用する環境。この環境には、Process Server Pod のいくつかのグループが含まれます。Business Central Monitoring を使用してサービスをデプロイし、実行し、停止し、またそれらの実行を監視します。
2 種類の管理環境をデプロイできます。**自由形式**のサーバー環境では、最初に Business 2 種類の管理環境をデプロイすることができます。自由形式のサーバー環境では、最初に Business Central Monitoring と 1 つの Process Server をデプロイします。その後、追加として任意の数の KIE Server をデプロイできます。この環境のデプロイメント手順については、[『Red Hat OpenShift Container Platform への Red Hat Process Automation Manager フリーフォーム管理サーバー環境のデプロイ』](#)を参照してください。

または、**固定**の管理サーバー環境をデプロイすることもできます。単一デプロイメントには、Business Central Monitoring、Smart Router、および事前に設定された数の Process Server (デフォルトでは 2 サーバーですが、テンプレートを変更して数を変更することができます) が含まれます。後のプロセスでは、サーバーを簡単に追加または削除することはできません。この環境のデプロイメント手順については、[『Red Hat OpenShift Container Platform への Red Hat Process Automation Manager 固定管理サーバー環境のデプロイ』](#)を参照してください。

- **イミュータブルサーバーを使用するデプロイメント:** 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代替の環境です。この環境では、Process Server Pod のデプロイ時にサービスまたはサービスのグループを読み込み、起動するイメージがビルドされます。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、Process Server は OpenShift 環境の Pod のように実行されるので、任意のコンテナベースの統合ワークフローを使用することができ、他のツールを使用して Pod を管理する必要はありません。オプションとして、Business Central Monitoring を使用して環境のパフォーマンスを監視できますが、追加のサービスを Process Server にデプロイしたり、既存のサービスのデプロイを解除したりすることはできません (コンテナの追加または削除はできません)。この環境のデプロイメント手順については、[『Red Hat OpenShift Container Platform への Red Hat Process Automation Manager イミュータブルサーバー環境のデプロイ』](#)を参照してください。

試用 または評価環境をデプロイすることも可能です。この環境には、Business Central と Process Server が含まれます。この環境はすばやく設定でき、これを使用して、アセットの開発や実行を評価し、体験できます。ただし、この環境では永続ストレージを使用せず、この環境でのいずれの作業も保存されません。この環境のデプロイメント手順については、[『Red Hat OpenShift Container Platform への Red Hat Process Automation Manager 試用環境のデプロイ』](#)を参照してください。

OpenShift に Red Hat Process Automation Manager 環境をデプロイするには、Red Hat Process Automation Manager で提供されるテンプレートを使用できます。設定が環境に適したものになるようにテンプレートを変更できます。

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

OpenShift 環境に Red Hat Process Automation Manager をデプロイする前に、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、プロセスの新しいバージョン、または別のプロセス)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. イメージストリームとイメージレジストリーの可用性確認

Red Hat Process Automation Manager コンポーネントを Red Hat OpenShift Container Platform にデプロイするには、OpenShift が Red Hat レジストリーから適切なイメージをダウンロードできることを確認する必要があります。これらのイメージをダウンロードするために、OpenShift ではイメージの場所情報が含まれる **イメージストリーム** が必要になります。また、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。

OpenShift 環境のバージョンによっては、必要なイメージストリームが含まれている場合があります。イメージストリームが提供されているかどうかを確認する必要があります。デフォルトでイメージストリームが OpenShift に含まれている場合は、OpenShift インフラストラクチャーがレジストリー認証サーバー用に設定されているのであれば、使用できます。管理者は、OpenShift 環境のインストール時に、レジストリーの認証設定を完了する必要があります。

それ以外の方法として、レジストリー認証を独自のプロジェクトで設定し、イメージストリームをそのプロジェクトにインストールすることができます。

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 が Red Hat レジストリーへのアクセス用のユーザー名とパスワードで設定されている場合は、以下のコマンドを実行します。

```
$ oc get imagestreamtag -n openshift | grep rhpam74-businesscentral
$ oc get imagestreamtag -n openshift | grep rhpam74-kieserver
```

両コマンドの出力が空でない場合は、必要なイメージストリームが **openshift** namespace にあるため、これ以外の操作は必要ありません。

3. コマンドの1つまたは複数の出力が空白の場合や、Red Hat レジストリーにアクセスするために、OpenShift をユーザー名およびパスワードで設定していない場合は、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、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する必要があります。
 - 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。

- f. [Software Downloads](#) ページから **rhpmam-7.4.0-openshift-templates.zip** の製品配信可能ファイルをダウンロードし、**rhpmam74-image-streams.yaml** ファイルを展開します。
- g. 以下のアクションの1つを完了します。
- 以下のコマンドを入力します。

```
$ oc create -f rhpmam74-image-streams.yaml
```

- OpenShift Web UI で、**Add to Project** → **Import YAML / JSON**を選択してからファイルを選択するか、またはその内容を貼り付けます。



注記

上記の手順を完了したら、イメージストリームを独自のプロジェクトの名前空間にインストールします。今回の例では、テンプレートのデプロイ時に **IMAGE_STREAM_NAMESPACE** パラメーターをこのプロジェクトの名前に設定する必要があります。

2.2. PROCESS 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。

Process Server への HTTP アクセス用に SSL 証明書を作成し、これをシークレットとして OpenShift 環境に指定する必要があります。

手順

- Process 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。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。



注記

実稼働環境で、Process Server の予想される URL と一致する有効な署名済み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。
- 証明書の名前をメモします。Red Hat Process Automation Manager 設定におけるこのデフォルト名は **jboss** です。

4. キーストアファイルのパスワードをメモします。Red Hat Process Automation Manager 設定におけるこのデフォルトの値は **mykeystorepass** です。
5. **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

Business Central への HTTP アクセス用に SSL 証明書を作成し、これをシークレットとして OpenShift 環境に指定する必要があります。

Business Central と Process 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。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。



注記

実稼働環境で、Business Central の予想される URL と一致する有効な署名済み証明書を生成します。

2. キーストアを **keystore.jks** ファイルに保存します。
3. 証明書の名前をメモします。Red Hat Process Automation Manager 設定におけるこのデフォルト名は **jboss** です。
4. キーストアファイルのパスワードをメモします。Red Hat Process Automation Manager 設定におけるこのデフォルトの値は **mykeystorepass** です。
5. **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 **businesscentral-app-secret** を生成します。

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

2.4. GLUSTERFS 設定の変更

OpenShift 環境が GlusterFS を使用して永続ストレージボリュームを提供するかどうかを確認する必要があります。GlusterFS を使用している場合は、最適なパフォーマンスを確保するために、ストレージクラスの設定を変更して GlusterFS ストレージをチューニングする必要があります。

手順

1. お使いの環境で GlusterFS が使用されているかどうかを確認するには、以下のコマンドを実行します。

```
oc get storageclass
```


この結果で、**(default)** マーカーが、**glusterfs** をリストするストレージクラスにあるかどうかを確認します。たとえば、以下の結果では、デフォルトのストレージクラスが **gluster-container** であり、**glusterfs** をリストします。

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

結果に、**glusterfs** をリストしないデフォルトストレージクラスが含まれる場合、または結果が空の場合は、変更する必要がありません。変更しない場合は、残りの手順を省略します。

2. デフォルトストレージクラスの設定を YAML ファイルに保存するには、以下のコマンドを実行します。

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

<class-name> はデフォルトのストレージクラス名に置き換えます。たとえば、以下のようになります。

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

3. **storage_config.yaml** ファイルを編集します。

- a. 以下のキーがある行を削除します。

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

- b. **volumeoptions** キーのある行に、**features.cache-invalidation on** および **performance.nl-cache on** の 2 つのパラメーターを追加します。以下に例を示します。

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

4. 既存のデフォルトストレージクラスを削除するには、以下のコマンドを実行します。

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

<class-name> はデフォルトのストレージクラス名に置き換えます。たとえば、以下のようになります。

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

5. 新しい設定を使用してストレージクラスを再作成するには、以下のコマンドを実行します。

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

2.5. 外部データベースのカスタム PROCESS SERVER 拡張イメージのビルド

Process Server に外部データベースサーバーを使用し、そのデータベースサーバーが MySQL または PostgreSQL 以外の場合は、環境をデプロイする前にこのサーバー用のドライバーを使用するカスタムの Process Server 拡張イメージをビルドする必要があります。

このビルド手順の手順を完了して、次のデータベースサーバーのいずれかにドライバーを提供します。

- Microsoft SQL Server
- MariaDB
- IBM DB2
- Oracle データベース
- Sybase

データベースサーバーのサポートされるバージョンについては、[Red Hat Process Automation Manager 7 でサポートされる設定](#) を参照してください。

ビルド手順では、既存の Process Server イメージを拡張するカスタム拡張イメージを作成します。このカスタム拡張イメージは OpenShift 環境にインポートしてから、**EXTENSIONS_IMAGE** パラメーターで参照する必要があります。

前提条件

- **oc** コマンドを使用して OpenShift 環境にログインしている。OpenShift ユーザーには **registry-editor** ロールが必要です。
- Oracle Database または Sybase の場合は、データベースサーバーベンダーから JDBC ドライバーをダウンロードしている。
- 以下の必要なソフトウェアをインストールしている。
 - Docker
 - Cekit バージョン 3.2
 - Cekit の以下のライブラリーおよび拡張機能:
 - **odcs-client**: **python3-odcs-client** パッケージまたは同様のパッケージで提供される。
 - **docker**: **python3-docker** パッケージまたは同様のパッケージで提供される。
 - **docker-squash**: **python3-docker-squash** または同様のパッケージで提供される。
 - **behave**: **python3-behave** パッケージまたは同様のパッケージで提供される。
 - **s2i**: **source-to-image** パッケージまたは同様のパッケージで提供される。

手順

1. IBM DB2、Oracle Database、または Sybase の場合、JDBC ドライバー JAR ファイルをローカルディレクトリーに指定します。

- Red Hat カスタマーポータルでの [Software Downloads](#) ページから利用可能な **rhpm-7.4.0-openshift-templates.zip** の製品配信可能ファイルをダウンロードします。
- ファイルを展開し、コマンドラインで、展開したファイルの **templates/contrib/jdbc** ディレクトリに変更します。このディレクトリには、カスタムビルドのソースコードが含まれません。
- データベースサーバーのタイプに応じて、以下のコマンドのいずれかを実行します。

- Microsoft SQL Server の場合:

```
make build mssql
```

- MariaDB の場合:

```
make build mariadb
```

- IBM DB2 の場合:

```
make build db2
```

- Oracle Database の場合:

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

このコマンドで、**/tmp/ojdbc7.jar** をダウンロードされた Oracle Database ドライバーのパス名に置き換え、**7.0** をドライバーのバージョンに置き換えます。

- Sybase の場合:

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

このコマンドで、**/tmp/jconn4-16.0_PL05.jar** をダウンロードされた Sybase ドライバーのパス名に置き換え、**16.0_PL05** をドライバーのバージョンに置き換えます。

- 以下のコマンドを実行して、ローカルで利用可能な Docker イメージを一覧表示します。

```
docker images
```

ビルドされたイメージの名前 (例: **jboss-kie-db2-extension-openshift-image**) およびイメージのバージョンタグ (**11.1.4.4** など (**latest** タグではない)) をメモします。

- OpenShift 環境のレジストリーに直接アクセスし、イメージをレジストリーにプッシュします。ユーザーパーミッションに応じて、イメージを **openshift** 名前空間またはプロジェクト名前空間にプッシュできます。レジストリーへのアクセスおよびイメージのプッシュの手順については、「[Accessing the Registry Directly](#)」を参照してください。
- 外部データベースサーバーをサポートするテンプレートを使って Process Server デプロイメントを設定する場合、以下のパラメーターを設定します。
 - Drivers Extension Image (EXTENSIONS_IMAGE)**: 拡張イメージの ImageStreamTag 定義 (例: **jboss-kie-db2-extension-openshift-image:11.1.4.4**)
 - Drivers ImageStream Namespace (EXTENSIONS_IMAGE_NAMESPACE)**: 拡張イメージのアップロード先の名前空間 (例: **openshift** またはプロジェクト名前空間)

2.6. オフラインで使用する MAVEN ミラーリポジトリーの用意

Red Hat OpenShift Container Platform 環境に公開インターネットへの送信アクセスが設定されていない場合には、必要なアーティファクトすべてのミラーが含まれる Maven リポジトリーを用意して、このリポジトリーを使用できるようにする必要があります。



注記

Red Hat OpenShift Container Platform 環境がインターネットに接続されている場合は、この手順を飛ばして次に進むことができます。

前提条件

- 公開インターネットへの送信アクセスが設定されているコンピューターが利用できる。

手順

- 書き込み可能な Maven リリースリポジトリーを準備します。このリポジトリーは、認証なしに読み込みアクセスを許可する必要があります。OpenShift 環境は、このリポジトリーへのアクセスが必要です。OpenShift 環境に、Nexus リポジトリーマネージャーをデプロイできます。OpenShift への Nexus の設定方法は、「Nexus の [設定](#)」を参照してください。このリポジトリーを別個のミラーリポジトリーとして使用します。
または、サービスにカスタムの外部リポジトリー (Nexus など) を使用する場合、同じリポジトリーをミラーリポジトリーとして使用できます。
- 公開インターネットに送信アクセスができるコンピューターで、以下のアクションを実行します。
 - 最新バージョンの [Offliner ツール](#) をダウンロードします。
 - Red Hat カスタマーポータル [Software Downloads](#) ページから利用可能な **rhpam-7.4.0-offliner.txt** の製品配信可能ファイルをダウンロードします。
 - 以下のコマンドを入力して、Offliner ツールを使用し、必要なアーティファクトをダウンロードします。

```
java -jar offliner-<version>.jar -r https://maven.repository.redhat.com/ga/ -r
https://repo1.maven.org/maven2/ -d /home/user/temp rhpam-7.4.0-offliner.txt
```

/home/user/temp は空の一時ディレクトリーに、**<version>** はダウンロードした Offliner ツールのバージョンに置き換えます。ダウンロードにはかなり時間がかかる可能性があります。

- このツールにより、ダウンロードの失敗が報告された場合には、以下のコマンドを入力して初回にダウンロードに失敗したアーティファクトをダウンロードします。

```
grep Path: errors.log | sed -n -e 's/^.*Path: //p' > failed-downloads.txt
java -jar offliner-<version>.jar -r https://maven.repository.redhat.com/ga/ -r
https://repo1.maven.org/maven2/ -d /home/user/temp failed-downloads.txt
```

もう一度ダウンロードの失敗が報告され、失敗数が初回のダウンロード合計数と比較して少しの場合には、続行してください。

- e. 一時ディレクトリーから作成した Maven リポジトリーにすべてのアーティファクトをアップロードします。アーティファクトのアップロードには、[Maven Repository Provisioner](#) ユーティリティーを使用できます。
3. Business Central 外でサービスを開発し、追加の依存関係がある場合は、ミラーリポジトリーにその依存関係を追加します。サービスを Maven プロジェクトとして開発した場合は、以下の手順を使用し、これらの依存関係を自動的に用意します。公開インターネットへに送信接続できるコンピューターで、この手順を実行します。
 - a. ローカルの Maven キャッシュディレクトリー (`~/.m2/repository`) のバックアップを作成して、ディレクトリーを削除します。
 - b. **mvn clean install** コマンドを使用してプロジェクトのソースをビルドします。
 - c. すべてのプロジェクトで以下のコマンドを入力し、Maven を使用してプロジェクトで生成したすべてのアーティファクトのランタイムの依存関係をすべてダウンロードするようにします。

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

`/path/to/project/pom.xml` は、プロジェクトの **pom.xml** ファイルへの正しいパスに置き換えます。

- d. ローカルの Maven キャッシュディレクトリー (`~/.m2/repository`) から作成した Maven ミラーリポジトリーにすべてのアーティファクトをアップロードします。アーティファクトのアップロードには、[Maven Repository Provisioner](#) ユーティリティーを使用できます。

第3章 フリーフォーム管理サーバー環境

Process Server を実行する複数の異なる Pod で設定される、フリーフォームサーバー環境をデプロイできます。これらの Process Server は、ステージングまたは実稼働環境を目的として、さまざまなサービスを実行できます。必要に応じて、サーバーをいつでも追加および削除できます。

Business Central Monitoring と管理 Process Server 1 台をデプロイし、フリーフォーム管理サーバー環境のデプロイを開始します。Business Central Monitoring を使用して Process Server 上のサービスの実行を監視し、必要に応じて管理できます。この環境には Smart Router は含まれていません。

追加の管理 Process Server をデプロイすることもできます。各 Process Server は、必要に応じて個別にスケールできます。

管理 Process Server では、最初からサービスが読み込まれるわけではありません。Business Central Monitoring または Process Server の REST API を使用して、サーバーでプロセスをデプロイおよびデプロイ解除してください。

サーバーにデプロイするプロセス (KJAR ファイル) がある Maven リポジトリを提供する必要があります。統合プロセスは、必要なバージョンのプロセスが Maven リポジトリにアップロードされるようにする必要があります。開発環境で Business Central を使用してプロセスを作成し、Maven リポジトリにアップロードできます。

各 Process Server はデータベースサーバーを使用します。通常、データベースサーバーも Pod で実行されますが、外部データベースサーバーを使用するように Process Server を設定できます。

同じ namespace にイミュータブル Process Server をデプロイすることもできます。Business Central Monitoring を使用して、イミュータブルサーバーなど、その環境のすべての Process Server に関するモニタリング情報を確認できます。イミュータブル Process Server のデプロイ手順については、『[Deploying a Red Hat Process Automation Manager immutable server environment on Red Hat OpenShift Container Platform](#)』を参照してください。

3.1. フリーフォーム環境用の MONITORING および単一 PROCESS SERVER のデプロイ

フリーフォーム環境のデプロイを開始するには、Business Central Monitoring と、Pod で PostgreSQL データベースサーバーを使用する管理 Process Server を 1 台デプロイします。Process Server には、サービスは読み込まれません。Business Central Monitoring を使用して、サーバーでサービスをデプロイおよびデプロイ解除します。

その後、必要に応じて Process Server を追加できます。

3.1.1. Monitoring および単一の Process Server 用のテンプレート設定の開始

Business Central Monitoring と管理 Process Server 1 台をデプロイするには、`rhpm74-managed.yaml` テンプレートファイルを使用します。

手順

1. Red Hat カスタマーポータル [の Software Downloads](#) ページから利用可能な `rhpm-7.4.0-openshift-templates.zip` の製品配信可能ファイルをダウンロードします。
2. `rhpm74-managed.yaml` テンプレートファイルを展開します。
3. 以下のいずれかの方法を使用してテンプレートのデプロイを開始します。

- OpenShift Web UI を使用するには、OpenShift アプリケーションコンソールで **Add to Project → Import YAML / JSON** を選択してから、**rhpm74-managed.yaml** ファイルを選択するか、またはこれを貼り付けます。Add Template ウィンドウで、**Process the template** が選択されていることを確認し、**Continue** をクリックします。
- OpenShift コマンドラインコンソールを使用するには、以下のコマンドラインを準備します。

```
oc new-app -f <template-path>/rhpm74-managed.yaml -p
BUSINESS_CENTRAL_HTTPS_SECRET=businesscentral-app-secret -p
KIE_SERVER_HTTPS_SECRET=kieserver-app-secret -p PARAMETER=value
```

このコマンドラインで、以下のように変更します。

- **<template-path>** を、ダウンロードしたテンプレートファイルのパスに置き換えます。
- 必要なパラメーターに設定するために必要な数だけ **-p PARAMETER=value** ペアを使用します。

次のステップ

テンプレートのパラメーターを設定します。「[Monitoring および単一の Process Server に必要なパラメーターの設定](#)」の手順を実行し、共通のパラメーターを設定します。テンプレートファイルを表示して、すべてのパラメーターの説明を確認します。

3.1.2. Monitoring および単一の Process Server に必要なパラメーターの設定

Business Central Monitoring と管理 Process Server 1 台をデプロイするようにテンプレートを設定するには、いずれの場合でも以下のパラメーターを設定する必要があります。

前提条件

- 「[Monitoring および単一の Process Server 用のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

1. 以下のパラメーターを設定します。
 - **Business Central Monitoring Server Keystore Secret Name (BUSINESS_CENTRAL_HTTPS_SECRET)**: 「[Business Central へのシークレットの作成](#)」で作成した Business Central のシークレットの名前。
 - **KIE Server Keystore Secret Name(KIE_SERVER_HTTPS_SECRET)**: 「[Process Server にシークレットの作成](#)」で作成した Process Server のシークレットの名前。
 - **Business Central Monitoring Server Certificate Name (BUSINESS_CENTRAL_HTTPS_NAME)**: 「[Business Central へのシークレットの作成](#)」で作成したキーストアの証明書の名前。
 - **Business Central Monitoring Server Keystore Password (BUSINESS_CENTRAL_HTTPS_PASSWORD)**: 「[Business Central へのシークレットの作成](#)」で作成したキーストアのパスワード。
 - **KIE Server Certificate Name(KIE_SERVER_HTTPS_NAME)**: 「[Process Server にシークレットの作成](#)」で作成したキーストアの証明書名。

- **KIE Server Keystore Password (KIE_SERVER_HTTPS_PASSWORD):** [「Process Server にシークレットの作成」](#) で作成したキーストアのパスワード。
 - **アプリケーション名 (APPLICATION_NAME):** OpenShift アプリケーションの名前。これは、Business Central Monitoring および Process Server のデフォルト URL で使用されます。OpenShift はアプリケーション名を使用して、デプロイメント設定、サービス、ルート、ラベル、およびアーティファクトの個別のセットを作成します。
 - **Maven repository URL (MAVEN_REPO_URL):** Maven リポジトリの URL。お使いの環境の Process Server にデプロイするすべてのプロセス (KJAR ファイル) をこのリポジトリにアップロードする必要があります。
 - **Maven リポジトリの ID (MAVEN_REPO_ID):** Maven リポジトリの ID。デフォルト値は **repo-custom** です。
 - **Maven repository username (MAVEN_REPO_USERNAME):** Maven リポジトリのユーザー名。
 - **Maven リポジトリのパスワード (MAVEN_REPO_PASSWORD):** Maven リポジトリのパスワード。
 - **KIE Server Mode (KIE_SERVER_MODE):** `rhcam74-managed.yaml` テンプレートで、デフォルト値は **PRODUCTION** です。**PRODUCTION** モードでは、**SNAPSHOT** バージョンの KJAR アーティファクトは Process Server にデプロイできず、既存のコンテナでアーティファクトのバージョンを変更できません。**PRODUCTION** モードで新規バージョンをデプロイするには、同じ Process Server で新規コンテナを作成します。**SNAPSHOT** バージョンをデプロイするか、または既存コンテナのアーティファクトのバージョンを変更するには、このパラメーターを **DEVELOPMENT** に設定します。
 - **ImageStream 名前空間 (IMAGE_STREAM_NAMESPACE):** イメージストリームが利用可能な名前空間。OpenShift 環境でイメージストリームがすでに利用可能な場合 ([「イメージストリームとイメージレジストリーの可用性確認」](#) を参照)、名前空間は **openshift** になります。イメージストリームファイルをインストールしている場合は、名前空間が OpenShift プロジェクトの名前になります。
2. 以下のユーザー名とパスワードを設定できます。デフォルトでは、デプロイメントはパスワードを自動的に生成します。
- **KIE Admin User (KIE_ADMIN_USER) および KIE Admin Password (KIE_ADMIN_PWD):** 管理者ユーザーのユーザー名およびパスワード。同じテンプレートでデプロイした Process Server 以外の Process Server を Business Central を使用して制御するか、または監視する場合には、ユーザー名およびパスワードを設定し、記録する必要があります。
 - **KIE Server User (KIE_SERVER_USER) および KIE Server Password (KIE_SERVER_PWD):** いずれかの Process Server に接続するのにクライアントアプリケーションが使用できるユーザー名およびパスワード。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、[「Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する」](#) の手順に従います。

3.1.3. Monitoring および単一の Process Server の Pod レプリカ数の設定

Business Central Monitoring および単一の管理 Process Server をデプロイするテンプレートを設定する場合には、Process Server および Business Central Monitoring のレプリカの初期数を設定します。

前提条件

- 「[Monitoring および単一の Process Server 用のテンプレート設定の開始](#)」 に説明されているようにテンプレートの設定を開始していること。

手順

レプリカの数を設定するには、次のパラメーターを設定します。

- **Business Central Monitoring Container Replicas** (**BUSINESS_CENTRAL_MONITORING_CONTAINER_REPLICAS**): デプロイメント時に最初に Business Central Monitoring 用に最初に作成されるレプリカ数。Business Central Monitoring に高可用性設定を使用しない場合は、この数値を 1 に設定します。
- **KIE Server Container Replicas**(**KIE_SERVER_CONTAINER_REPLICAS**): デプロイメントで Process Server 用に最初に作成されるレプリカの数。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する](#)」の手順に従います。

3.1.4. 公開インターネットへの接続のない環境で Monitoring および単一の Process Server の Maven ミラーへのアクセス設定

Business Central Monitoring および単一の管理 Process Server をデプロイするようにテンプレートを設定する時に、OpenShift 環境に公開インターネットへの接続がない場合は、「[オフラインで使用する Maven ミラーリポジトリーの用意](#)」に従って設定した Maven ミラーへのアクセスを設定する必要があります。

前提条件

- 「[Monitoring および単一の Process Server 用のテンプレート設定の開始](#)」 に説明されているようにテンプレートの設定を開始していること。

手順

Maven ミラーへのアクセスを設定するには、以下のパラメーターを設定します。

- **Maven mirror URL**(**MAVEN_MIRROR_URL**): 「[オフラインで使用する Maven ミラーリポジトリーの用意](#)」 で設定した Maven ミラーリポジトリーの URL。この URL は、OpenShift 環境の Pod からアクセスできるようにする必要があります。
- **Maven mirror of**(**MAVEN_MIRROR_OF**): ミラーから取得されるアーティファクトを定める値。**mirrorOf** 値の設定方法は、Apache Maven ドキュメントの [Mirror Settings](#) を参照してください。デフォルト値は **external:*** です。この値の場合、Maven はミラーから必要なアーティファクトをすべて取得し、他のリポジトリーにクエリーを送信しません。
 - 外部の Maven リポジトリー (**MAVEN_REPO_URL**) を設定する場合は、ミラーからこのリポジトリー内のアーティファクトを除外するように **MAVEN_MIRROR_OF** を変更します (例: **external:*;!repo-custom**)。 **repo-custom** は、**MAVEN_REPO_ID** で設定した ID に置き換えます。

- ビルトイン Business Central Maven リポジトリ (**BUSINESS_CENTRAL_MAVEN_SERVICE**) を設定する場合には、ミラーからこのリポジトリのアーティファクトを除外するように **MAVEN_MIRROR_OF** を変更します (例: **external:*,!repo-rhpamcentr**)。
- 両方のリポジトリを設定している場合は、ミラーからこのリポジトリを両方除外するように **MAVEN_MIRROR_OF** を変更します (例: **external:*,!repo-rhpamcentr,!repo-custom**)。 **repo-custom** は、**MAVEN_REPO_ID** で設定した ID に置き換えます。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する](#)」の手順に従います。

3.1.5. Monitoring および単一の Process Server の RH-SSO 認証用パラメーター設定

RH-SSO 認証を使用する場合は、Business Central Monitoring および単一の管理 Process Server をデプロイするようにテンプレートを設定するには次の追加の設定を実行します。



重要

LDAP 認証および RH-SSO 認証を同じデプロイメントに設定しないようにしてください。

前提条件

- Red Hat Process Automation Manager のレムが RH-SSO 認証システムに作成されている。
- Red Hat Process Automation Manager のユーザー名およびパスワードが RH-SSO 認証システムに作成されている。利用可能なロールの一覧は、[4章 Red Hat Process Automation Manager ロールおよびユーザー](#) を参照してください。以下のユーザーは、環境のパラメーターを設定するために必要です。
 - **kie-server,rest-all,admin** ロールを持つ管理者ユーザー。このユーザーは環境を管理し、これを使用できます。Process Server はこのユーザーを使用して Business Central Monitoring で認証します。
 - **kie-server,rest-all,user** ロールを持つサーバーユーザー。このユーザーは、Process Server に対する REST API 呼び出しを実行できます。Business Central Monitoring はこのユーザーを使用して Process Server で認証します。
- クライアントが、デプロイしている Red Hat Process Automation Manager 環境のすべてのコンポーネントについて RH-SSO 認証システムに作成されている。クライアントのセットアップには、コンポーネントの URL が含まれます。環境のデプロイ後に URL を確認し、編集できます。または、Red Hat Process Automation Manager デプロイメントはクライアントを作成できます。ただし、このオプションの環境に対する制御の詳細度はより低くなります。
- 「[Monitoring および単一の Process Server 用のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

1. テンプレートの **KIE_ADMIN_USER** および **KIE_ADMIN_PASSWORD** パラメーターを、RH-SSO 認証システムで作成したユーザー名およびパスワードに設定します。

2. テンプレートの **KIE_SERVER_USER** および **KIE_SERVER_PASSWORD** パラメーターを、RH-SSO 認証システムで作成したサーバーユーザーのユーザー名およびパスワードに設定します。
3. 以下のパラメーターを設定します。
 - **RH-SSO URL (SSO_URL)**: RH-SSO の URL。
 - **RH-SSO Realm name (SSO_REALM)**: Red Hat Process Automation Manager の RH-SSO レルム。
 - **RH-SSO が無効な SSL 証明書の検証 (SSO_DISABLE_SSL_CERTIFICATE_VALIDATION)**: RH-SSO インストールで有効な HTTPS 証明書を使用していない場合は **true** に設定します。
4. 以下の手順のいずれかを実行します。
 - a. RH-SSO で Red Hat Process Automation Manager のクライアントを作成した場合は、テンプレートで以下のパラメーターを設定します。
 - **Business Central Monitoring RH-SSO Client name (BUSINESS_CENTRAL_SSO_CLIENT)**: Business Central Monitoring の RH-SSO クライアント名。
 - **Business Central Monitoring RH-SSO Client Secret (BUSINESS_CENTRAL_SSO_SECRET)**: Business Central Monitoring のクライアント向けに RH-SSO に設定されているシークレット文字列。
 - **KIE Server RH-SSO Client name (KIE_SERVER_SSO_CLIENT)**: Process Server の RH-SSO クライアント名。
 - **KIE Server RH-SSO Client Secret (KIE_SERVER_SSO_SECRET)**: Process Server のクライアントに対して RH-SSO に設定するシークレットの文字列。
 - b. RH-SSO で Red Hat Process Automation Manager のクライアントを作成するには、テンプレートで以下のパラメーターを設定します。
 - **Business Central Monitoring RH-SSO Client name (BUSINESS_CENTRAL_SSO_CLIENT)**: Business Central Monitoring の RH-SSO で作成するクライアントの名前。
 - **Business Central Monitoring RH-SSO Client Secret (BUSINESS_CENTRAL_SSO_SECRET)**: Business Central Monitoring のクライアント向けに RH-SSO に設定するシークレット文字列。
 - **KIE Server RH-SSO Client name (KIE_SERVER_SSO_CLIENT)**: Process Server 向けに RH-SSO に作成するクライアント名。
 - **KIE Server RH-SSO Client Secret (KIE_SERVER_SSO_SECRET)**: Process Server のクライアントに対して RH-SSO に設定するシークレットの文字列。
 - **RH-SSO Realm Admin Username (SSO_USERNAME)** および **RH-SSO Realm Admin Password (SSO_PASSWORD)**: Red Hat Process Automation Manager の RH-SSO レルムのレルム管理者ユーザーのユーザー名およびパスワード。必要なクライアントを作成するためにこのユーザー名およびパスワードを指定する必要があります。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する](#)」の手順に従います。

デプロイの完了後に、RH-SSO 認証システムで Red Hat Process Automation Manager のコンポーネントの URL が正しいことを確認してください。

3.1.6. Monitoring および単一の Process Server の LDAP 認証用パラメーター設定

LDAP 認証を使用する場合は、Business Central Monitoring および単一の管理 Process Server をデプロイするようにテンプレートを設定する時に、次の追加の設定を実行します。



重要

LDAP 認証および RH-SSO 認証を同じデプロイメントに設定しないようにしてください。

前提条件

- LDAP システムに Red Hat Process Automation Manager のユーザー名およびパスワードを作成していること。利用可能なロールの一覧は、[4章 Red Hat Process Automation Manager ロールおよびユーザー](#)を参照してください。この環境のパラメーターを設定するために、少なくとも以下のユーザーを作成している必要があります。
 - **kie-server,rest-all,admin** ロールを持つ管理者ユーザー。このユーザーは環境を管理し、これを使用できます。
 - **kie-server,rest-all,user** ロールを持つサーバーユーザー。このユーザーは、Process Server に対する REST API 呼び出しを実行できます。
- 「[Monitoring および単一の Process Server 用のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

1. LDAP サービスでは、デプロイメントパラメーターですべてのユーザー名を作成します。パラメーターを設定しない場合には、デフォルトのユーザー名を使用してユーザーを作成します。作成したユーザーにはロールに割り当てる必要もあります。
 - **KIE_ADMIN_USER**: デフォルトのユーザー名 **adminUser**、ロール: **kie-server,rest-all,admin**
 - **KIE_SERVER_USER**: デフォルトのユーザー名 **executionUser**、ロール **kie-server,rest-all,guest**
LDAP で設定可能なユーザーロールについては、「[ロール およびユーザー](#)」を参照してください。
2. テンプレートの **AUTH_LDAP*** パラメーターを設定します。これらのパラメーターは、Red Hat JBoss EAP の **LdapExtended** ログインモジュールの設定に対応します。これらの設定に関する説明は、[LdapExtended login module](#) を参照してください。
LDAP サーバーでデプロイメントに必要なすべてのロールが定義されていない場合は、Red Hat Process Automation Manager ロールに LDAP グループをマップできます。LDAP のロールマッピングを有効にするには、以下のパラメーターを設定します。
 - **RoleMapping rolesProperties** ファイルパス

(AUTH_ROLE_MAPPER_ROLES_PROPERTIES):

`/opt/eap/standalone/configuration/rolemapping/rolemapping.properties` など、ロールのマッピングを定義するファイルの完全修飾パス名。このファイルを指定して、該当するすべてのデプロイメント設定でこのパスにマウントする必要があります。これを実行する方法については、「[\(任意\) 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。

- **RoleMapping replaceRole** プロパティ (**AUTH_ROLE_MAPPER_REPLACE_ROLE**): **true** に設定した場合、マッピングしたロールは、LDAP サーバーに定義したロールに置き換えられます。**false** に設定した場合は、LDAP サーバーに定義したロールと、マッピングしたロールの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。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する](#)」の手順に従います。

3.1.7. Monitoring および単一の Process Server の Prometheus メトリクス収集の有効化

Process Server デプロイメントを Prometheus を使用してメトリクスを収集し、保存するように設定する必要がある場合、デプロイ時に Process Server でこの機能のサポートを有効にします。

前提条件

- 「[Monitoring および単一の Process Server 用のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

Prometheus メトリクス収集のサポートを有効にするには、**Prometheus Server 拡張無効 (PROMETHEUS_SERVER_EXT_DISABLED)** パラメーターを **false** に設定します。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する](#)」の手順に従います。

Prometheus メトリクス収集の方法については、「[Process Server の管理および監視](#)」を参照してください。

3.1.8. Monitoring および単一の Process Server 用のテンプレートのデプロイメントを完了する

OpenShift Web UI またはコマンドラインで必要なすべてのパラメーターを設定した後に、テンプレートのデプロイを実行します。

手順

使用している方法に応じて、以下の手順を実行します。

- OpenShift Web UI の場合は **Create** をクリックします。

- **This will create resources that may have security or project behavior implications**
メッセージが表示された場合は、**Create Anyway** をクリックします。
- コマンドラインに入力して、Enter キーを押します。

3.2. フリーフォーム環境用の追加の管理 PROCESS SERVER のデプロイ

管理 Process Server は、フリーフォーム環境に追加できます。このサーバーは、Pod で PostgreSQL または MySQL データベースサーバーを使用するか、または外部データベースサーバーを使用できません。

Business Central Monitoring デプロイメントと同じプロジェクトにサーバーをデプロイします。

Process Server は、Maven リポジトリからサービスを読み込みます。

サーバーは、サービスが読み込まれていない状態で起動します。サーバーへのサービスのデプロイおよびデプロイの解除は、Business Central Monitoring または Process Server の REST API を使用します。

3.2.1. 追加の管理 Process Server のテンプレート設定の開始

追加の管理 Process Server をデプロイするには、以下のテンプレートファイルのいずれかを使用します。

- **rhpm74-kieserver-postgresql.yaml**: 永続ストレージに PostgreSQL Pod を使用します。他のテンプレートを使用する特別な理由がない限り、このテンプレートを使用します。
- **rhpm74-kieserver-mysql.yaml**: 永続ストレージに MySQL Pod を使用します。
- **rhpm74-kieserver-externaldb.yaml**: 永続ストレージに外部データベースサーバーを使用します。



重要

外部データベースサーバーの標準 Process Server イメージには、MySQL および PostgreSQL 外部データベースサーバー用のドライバーが含まれます。別のデータベースサーバーを使用する場合は、カスタムの Process Server イメージをビルドする必要があります。手順は、「[外部データベースのカスタム Process Server 拡張イメージのビルド](#)」を参照してください。

手順

1. Red Hat カスタマーポータルでの [Software Downloads](#) ページから利用可能な **rhpm-7.4.0-openshift-templates.zip** の製品配信可能ファイルをダウンロードします。
2. 必要なテンプレートファイルを展開します。
3. 以下のいずれかの方法を使用してテンプレートのデプロイを開始します。
 - OpenShift Web UI を使用するには、OpenShift アプリケーションコンソールで **Add to Project → Import YAML / JSON** を選択してから **<template-file-name>.yaml** ファイルを選択または貼り付けます。Add Template ウィンドウで、**Process the template** が選択されていることを確認し、**Continue** をクリックします。
 - OpenShift コマンドラインコンソールを使用するには、以下のコマンドラインを準備します。
 -

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

このコマンドラインで、以下のように変更します。

- **<template-path>** を、ダウンロードしたテンプレートファイルのパスに置き換えます。
- **<template-file-name>** は、テンプレート名に置き換えます。
- 必要なパラメーターに設定するために必要な数だけ **-p PARAMETER=value** ペアを使用します。

次のステップ

テンプレートのパラメーターを設定します。「[追加の管理 Process Server に必要なパラメーターの設定](#)」の手順に従い、共通のパラメーターを設定します。テンプレートファイルを表示して、すべてのパラメーターの説明を確認します。

3.2.2. 追加の管理 Process Server に必要なパラメーターの設定

テンプレートを追加の管理 Process Server をデプロイするように設定する場合は、常に以下のパラメーターを設定する必要があります。

前提条件

- 「[追加の管理 Process Server のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

1. 以下のパラメーターを設定します。
 - **KIE Server Keystore Secret Name(KIE_SERVER_HTTPS_SECRET)**: 「[Process Server にシークレットの作成](#)」で作成した Process Server のシークレットの名前。
 - **KIE Server Certificate Name(KIE_SERVER_HTTPS_NAME)**: 「[Process Server にシークレットの作成](#)」で作成したキーストアの証明書名。
 - **KIE Server Keystore Password(KIE_SERVER_HTTPS_PASSWORD)**: 「[Process Server にシークレットの作成](#)」で作成したキーストアのパスワード。
 - **アプリケーション名 (APPLICATION_NAME)**: OpenShift アプリケーションの名前。これは、Business Central Monitoring および Process Server のデフォルト URL で使用されます。OpenShift はアプリケーション名を使用して、デプロイメント設定、サービス、ルート、ラベル、およびアーティファクトの個別のセットを作成します。同じテンプレートを同じプロジェクトで使用して複数のアプリケーションをデプロイすることもできますが、その場合はアプリケーション名を同じにすることはできません。また、アプリケーション名は、Process Server が Business Central Monitoring で参加するサーバーの設定 (サーバーテンプレート) の名前を決定するものとなります。複数の Process Server をデプロイしている場合、それぞれのサーバーに異なるアプリケーション名があることを確認する必要があります。
 - **Maven repository URL(MAVEN_REPO_URL)**: Maven リポジトリーの URL。Process Server にデプロイするすべてのプロセス (KJAR ファイル) をこのリポジトリーにアップロードする必要があります。

- **Maven リポジトリーの ID (MAVEN_REPO_ID):** Maven リポジトリーの ID。デフォルト値は **repo-custom** です。
 - **Maven repository username (MAVEN_REPO_USERNAME):** Maven リポジトリーのユーザー名。
 - **Maven リポジトリーのパスワード (MAVEN_REPO_PASSWORD):** Maven リポジトリーのパスワード。
 - **KIE Server Mode (KIE_SERVER_MODE):** `rhpam74-kieserver-*.yaml` テンプレートで、デフォルト値は **PRODUCTION** です。 **PRODUCTION** モードでは、 **SNAPSHOT** バージョンの KJAR アーティファクトは Process Server にデプロイできず、既存のコンテナでアーティファクトのバージョンを変更できません。 **PRODUCTION** モードで新規バージョンをデプロイするには、同じ Process Server で新規コンテナを作成します。 **SNAPSHOT** バージョンをデプロイするか、または既存コンテナのアーティファクトのバージョンを変更するには、このパラメーターを **DEVELOPMENT** に設定します。
 - **ImageStream 名前空間 (IMAGE_STREAM_NAMESPACE):** イメージストリームが利用可能な名前空間。 OpenShift 環境でイメージストリームがすでに利用可能な場合（「[イメージストリームとイメージレジストリーの可用性確認](#)」を参照）、名前空間は **openshift** になります。 イメージストリームファイルをインストールしている場合は、名前空間が OpenShift プロジェクトの名前になります。
2. 以下のユーザー名とパスワードを設定できます。デフォルトでは、デプロイすると、パスワードが自動生成されます。
- **KIE Server User (KIE_SERVER_USER) および KIE Server Password (KIE_SERVER_PWD):** いずれかの Process Server に接続するのにクライアントアプリケーションが使用できるユーザー名およびパスワード。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[追加の管理 Process Server のテンプレートデプロイメントの開始](#)」の手順に従います。

3.2.3. 追加の管理 Process Server のイメージストリーム namespace の設定

openshift ではない名前空間でイメージストリームを作成した場合は、テンプレートで名前空間を設定する必要があります。

すべてのイメージストリームが Red Hat OpenShift Container Platform 環境ですでに利用可能な場合は、この手順を省略できます。

前提条件

- 「[追加の管理 Process Server のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

「[イメージストリームとイメージレジストリーの可用性確認](#)」の説明に従ってイメージストリームファイルをインストールした場合は、ImageStream 名前空間 (**IMAGE_STREAM_NAMESPACE**) パラメーターを OpenShift プロジェクトの名前に設定します。

3.2.4. 追加の管理 Process Server 用の Business Central Monitoring インスタンスについての情報の設定

デプロイした Business Central Monitoring インスタンスから Process Server への接続を有効にするには、Business Central Monitoring インスタンスの情報を設定する必要があります。

前提条件

- 「追加の管理 Process Server のテンプレート設定の開始」 に説明されているようにテンプレートの設定を開始していること。

手順

1. 以下のパラメーターを設定します。
 - **KIE Admin User (KIE_ADMIN_USER)** および **KIE Admin Password (KIE_ADMIN_PWD)**: 管理者ユーザーのユーザー名およびパスワード。これらの値は Business Central Monitoring の **KIE_ADMIN_USER** および **KIE_ADMIN_PWD** 設定と同じである必要があります。Business Central Monitoring が RH-SSO または LDAP 認証を使用する場合には、これらの値は Business Central Monitoring の管理者ロールを使用して認証システムに設定されるユーザー名およびパスワードである必要があります。
 - **Name of the Business Central service (BUSINESS_CENTRAL_SERVICE)**: Business Central Monitoring の OpenShift サービス名。
2. 以下の設定が Business Central Monitoring の同じ設定と値に設定されていることを確認します。
 - **Maven repository URL (MAVEN_REPO_URL)**: サービスのデプロイに使用する必要のある外部 Maven リポジトリーの URL。
 - **Maven repository username (MAVEN_REPO_USERNAME)**: Maven リポジトリーのユーザー名。
 - **Maven リポジトリーのパスワード (MAVEN_REPO_PASSWORD)**: Maven リポジトリーのパスワード。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「追加の管理 Process Server のテンプレートデプロイメントの開始」の手順に従います。

3.2.5. 追加の管理 Process Server の公開インターネットへの接続のない環境に Maven ミラーへのアクセスする手順

テンプレートを追加の管理 Process Server をデプロイするように設定する際に、OpenShift 環境に公開インターネットへの接続がない場合は、「オフラインで使用する Maven ミラーリポジトリーの用意」に従って設定した Maven ミラーへのアクセスを設定する必要があります。

前提条件

- 「追加の管理 Process Server のテンプレート設定の開始」 に説明されているようにテンプレートの設定を開始していること。

手順

Maven ミラーへのアクセスを設定するには、以下のパラメーターを設定します。

- **Maven mirror URL (MAVEN_MIRROR_URL)**: 「[オフラインで使用する Maven ミラーリポジトリー](#)」で設定した Maven ミラーリポジトリーの URL。この URL は、OpenShift 環境の Pod からアクセスできるようにする必要があります。
- **Maven mirror of (MAVEN_MIRROR_OF)**: ミラーから取得されるアーティファクトを定める値。**mirrorOf** 値の設定方法は、Apache Maven ドキュメントの [Mirror Settings](#) を参照してください。デフォルト値は **external:*** です。この値の場合、Maven はミラーから必要なアーティファクトをすべて取得し、他のリポジトリーにクエリーを送信しません。
 - 外部の Maven リポジトリー (**MAVEN_REPO_URL**) を設定する場合は、ミラーからこのリポジトリー内のアーティファクトを除外するように **MAVEN_MIRROR_OF** を変更します (例: **external:*;!repo-custom**)。repo-custom は、**MAVEN_REPO_ID** で設定した ID に置き換えます。
 - ビルトイン Business Central Maven リポジトリー (**BUSINESS_CENTRAL_MAVEN_SERVICE**) を設定する場合には、ミラーからこのリポジトリーのアーティファクトを除外するように **MAVEN_MIRROR_OF** を変更します (例: **external:*;!repo-rhpamcentr**)。
 - 両方のリポジトリーを設定している場合は、ミラーからこのリポジトリーを両方除外するように **MAVEN_MIRROR_OF** を変更します (例: **external:*;!repo-rhpamcentr;!repo-custom**)。repo-custom は、**MAVEN_REPO_ID** で設定した ID に置き換えます。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[追加の管理 Process Server のテンプレートデプロイメントの開始](#)」の手順に従います。

3.2.6. 追加の管理 Process Server の RH-SSO 認証パラメーターの設定

RH-SSO 認証を使用する場合は、追加の管理 Process Server をデプロイするようにテンプレートを設定する時に次の追加の設定を実行します。



重要

LDAP 認証および RH-SSO 認証を同じデプロイメントに設定しないようにしてください。

前提条件

- Red Hat Process Automation Manager のレلمが RH-SSO 認証システムに作成されている。
- Red Hat Process Automation Manager のユーザー名およびパスワードが RH-SSO 認証システムに作成されている。利用可能なロールの一覧は、[4章 Red Hat Process Automation Manager ロールおよびユーザー](#) を参照してください。以下のユーザーは、環境のパラメーターを設定するために必要です。
 - **kie-server,rest-all,admin** ロールを持つ管理者ユーザー。このユーザーは環境を管理し、これを使用できます。Process Server はこのユーザーを使用して Business Central Monitoring で認証します。

- **kie-server,rest-all,user** ロールを持つサーバーユーザー。このユーザーは、Process Server に対する REST API 呼び出しを実行できます。Business Central Monitoring はこのユーザーを使用して Process Server で認証します。
- クライアントが、デプロイしている Red Hat Process Automation Manager 環境のすべてのコンポーネントについて RH-SSO 認証システムに作成されている。クライアントのセットアップには、コンポーネントの URL が含まれます。環境のデプロイ後に URL を確認し、編集できます。または、Red Hat Process Automation Manager デプロイメントはクライアントを作成できます。ただし、このオプションの環境に対する制御の詳細度合はより低くなります。
- 「追加の管理 Process Server のテンプレート設定の開始」 に説明されているようにテンプレートの設定を開始していること。

手順

1. テンプレートの **KIE_ADMIN_USER** および **KIE_ADMIN_PASSWORD** パラメーターを、RH-SSO 認証システムで作成したユーザー名およびパスワードに設定します。
2. テンプレートの **KIE_SERVER_USER** および **KIE_SERVER_PASSWORD** パラメーターを、RH-SSO 認証システムで作成したサーバーユーザーのユーザー名およびパスワードに設定します。
3. 以下のパラメーターを設定します。
 - **RH-SSO URL (SSO_URL)**: RH-SSO の URL。
 - **RH-SSO Realm name (SSO_REALM)**: Red Hat Process Automation Manager の RH-SSO レルム。
 - **RH-SSO が無効な SSL 証明書の検証 (SSO_DISABLE_SSL_CERTIFICATE_VALIDATION)**: RH-SSO インストールで有効な HTTPS 証明書を使用していない場合は **true** に設定します。
4. 以下の手順のいずれかを実行します。
 - a. RH-SSO で Red Hat Process Automation Manager のクライアントを作成した場合は、テンプレートで以下のパラメーターを設定します。
 - **Business Central Monitoring RH-SSO Client name (BUSINESS_CENTRAL_SSO_CLIENT)**: Business Central Monitoring の RH-SSO クライアント名。
 - **KIE Server RH-SSO Client name (KIE_SERVER_SSO_CLIENT)**: Process Server の RH-SSO クライアント名。
 - **KIE Server RH-SSO Client Secret (KIE_SERVER_SSO_SECRET)**: Process Server のクライアントに対して RH-SSO に設定するシークレットの文字列。
 - b. RH-SSO で Red Hat Process Automation Manager のクライアントを作成するには、テンプレートで以下のパラメーターを設定します。
 - **KIE Server RH-SSO Client name (KIE_SERVER_SSO_CLIENT)**: Process Server 向けに RH-SSO に作成するクライアント名。
 - **KIE Server RH-SSO Client Secret (KIE_SERVER_SSO_SECRET)**: Process Server のクライアントに対して RH-SSO に設定するシークレットの文字列。
 - **RH-SSO Realm Admin Username (SSO_USERNAME)** および **RH-SSO Realm Admin**

Password (SSO_PASSWORD): Red Hat Process Automation Manager の RH-SSO レルムのレルム管理者ユーザーのユーザー名およびパスワード。必要なクライアントを作成するためにこのユーザー名およびパスワードを指定する必要があります。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[追加の管理 Process Server のテンプレートデプロイメントの開始](#)」の手順に従います。

デプロイの完了後に、RH-SSO 認証システムで Red Hat Process Automation Manager のコンポーネントの URL が正しいことを確認してください。

3.2.7. 追加の管理 Process Server の LDAP 認証パラメーターの設定

LDAP 認証を使用する場合は、追加の管理 Process Server をデプロイするようにテンプレートを設定する時に次の追加の設定を実行します。



重要

LDAP 認証および RH-SSO 認証を同じデプロイメントに設定しないようにしてください。

前提条件

- LDAP システムに Red Hat Process Automation Manager のユーザー名およびパスワードを作成していること。利用可能なロールの一覧は、[4章 Red Hat Process Automation Manager ロールおよびユーザー](#)を参照してください。この環境のパラメーターを設定するために、少なくとも以下のユーザーを作成している必要があります。
 - **kie-server,rest-all,admin** ロールを持つ管理者ユーザー。このユーザーは環境を管理し、これを使用できます。
 - **kie-server,rest-all,user** ロールを持つサーバーユーザー。このユーザーは、Process Server に対する REST API 呼び出しを実行できます。
- 「[追加の管理 Process Server のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

1. LDAP サービスでは、デプロイメントパラメーターですべてのユーザー名を作成します。パラメーターを設定しない場合には、デフォルトのユーザー名を使用してユーザーを作成します。作成したユーザーにはロールに割り当てる必要もあります。
 - **KIE_ADMIN_USER:** デフォルトのユーザー名 **adminUser**、ロール: **kie-server,rest-all,admin**
 - **KIE_SERVER_USER:** デフォルトのユーザー名 **executionUser**、ロール **kie-server,rest-all,guest**
LDAP で設定可能なユーザーロールについては、「[ロール およびユーザー](#)」を参照してください。

2. テンプレートの **AUTH_LDAP*** パラメーターを設定します。これらのパラメーターは、Red Hat JBoss EAP の **LdapExtended** ログインモジュールの設定に対応します。これらの設定に関する説明は、[LdapExtended login module](#) を参照してください。
LDAP サーバーでデプロイメントに必要なすべてのロールが定義されていない場合は、Red Hat Process Automation Manager ロールに LDAP グループをマップできます。LDAP のロールマッピングを有効にするには、以下のパラメーターを設定します。

- **RoleMapping rolesProperties** ファイルパス (**AUTH_ROLE_MAPPER_ROLES_PROPERTIES**):
`/opt/eap/standalone/configuration/rolemapping/rolemapping.properties` など、ロールのマッピングを定義するファイルの完全修飾パス名。このファイルを指定して、該当するすべてのデプロイメント設定でこのパスにマウントする必要があります。これを実行する方法については、「[\(任意\) 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。
- **RoleMapping replaceRole** プロパティ (**AUTH_ROLE_MAPPER_REPLACE_ROLE**):
true に設定した場合、マッピングしたロールは、LDAP サーバーに定義したロールに置き換えられます。**false** に設定した場合は、LDAP サーバーに定義したロールと、マッピングしたロールの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。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[追加の管理 Process Server のテンプレートデプロイメントの開始](#)」の手順に従います。

3.2.8. 追加の管理 Process Server 用に外部データベースサーバーを使用するためのパラメーターの設定

rhpbam74-kieserver-externaldb.yaml テンプレートを使用して Process Server 用に外部データベースサーバーを使用する場合は、追加の管理 Process Server をデプロイするようにテンプレートを設定する時に次の追加の設定を実行します。

前提条件

- 「[追加の管理 Process Server のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

1. 以下のパラメーターを設定します。
 - **KIE Server External Database Driver**(**KIE_SERVER_EXTERNALDB_DRIVER**): サーバーの種類に応じたサーバーのドライバー。
 - **mysql**
 - **postgresql**
 - **mariadb**
 - **mssql**
 - **db2**

- **oracle**
 - **sybase**
 - **KIE Server External Database User(KIE_SERVER_EXTERNALDB_USER)** および **KIE Server External Database Password (KIE_SERVER_EXTERNALDB_PWD)**: 外部データベースサーバーのユーザー名およびパスワード。
 - **KIE Server External Database URL(KIE_SERVER_EXTERNALDB_URL)**: 外部データベースサーバーの JDBC URL。
 - **KIE Server External Database Dialect(KIE_SERVER_EXTERNALDB_DIALECT)**: サーバーの種類に応じたサーバーの Hibernate ダイアレクト。
 - **org.hibernate.dialect.MySQL5InnoDBDialect** (MySQL および MariaDB で使用される)
 - **org.hibernate.dialect.PostgreSQL82Dialect**
 - **org.hibernate.dialect.SQLServer2012Dialect** (MS SQL で使用される)
 - **org.hibernate.dialect.DB2Dialect**
 - **org.hibernate.dialect.Oracle10gDialect**
 - **org.hibernate.dialect.SybaseASE157Dialect**
 - **KIE Server External Database Host(KIE_SERVER_EXTERNALDB_SERVICE_HOST)**: 外部データベースサーバーのホスト名。
 - **KIE Server External Database Port(KIE_SERVER_EXTERNALDB_SERVICE_PORT)**: 外部データベースサーバーのポート番号。
 - **KIE Server External Database name(KIE_SERVER_EXTERNALDB_DB)**: 外部データベースサーバーで使用するデータベース名。
 - **JDBC Connection Checker class (KIE_SERVER_EXTERNALDB_CONNECTION_CHECKER)**: データベースサーバーの JDBC connection checker class の名前。この情報がないと、データベースサーバー接続は、データベースサーバーの再起動時などで接続が失われた後に復元することができません。
 - **JDBC Exception Sorter class (KIE_SERVER_EXTERNALDB_EXCEPTION_SORTER)**: データベースサーバーの JDBC exception sorter class の名前。この情報がないと、データベースサーバー接続は、データベースサーバーの再起動時などで接続が失われた後に復元することができません。
2. 「外部データベースのカスタム Process Server 拡張イメージのビルド」で説明されているように、MySQL または PostgreSQL 以外の外部データベースサーバーを使用するためにカスタムイメージを作成している場合は、以下のパラメーターを設定します。
- **Drivers Extension Image (EXTENSIONS_IMAGE)**: 拡張イメージの ImageStreamTag 定義 (例: **jboss-kie-db2-extension-openshift-image:11.1.4.4**)
 - **Drivers ImageStream Namespace (EXTENSIONS_IMAGE_NAMESPACE)**: 拡張イメージのアップロード先の名前空間 (例: **openshift** またはプロジェクト名前空間)

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[追加の管理 Process Server のテンプレートデプロイメントの開始](#)」の手順に従います。

3.2.9. 追加の管理 Process Server の Prometheus メトリクス収集の有効化

Process Server デプロイメントを Prometheus を使用してメトリクスを収集し、保存するように設定する必要がある場合、デプロイ時に Process Server でこの機能のサポートを有効にします。

前提条件

- 「[追加の管理 Process Server のテンプレート設定の開始](#)」に説明されているようにテンプレートの設定を開始していること。

手順

Prometheus メトリクス収集のサポートを有効にするには、**Prometheus Server 拡張無効 (PROMETHEUS_SERVER_EXT_DISABLED)** パラメーターを **false** に設定します。

次のステップ

必要な場合は、追加のパラメーターを設定します。

デプロイを完了するには、「[追加の管理 Process Server のテンプレートデプロイメントの開始](#)」の手順に従います。

Prometheus メトリクス収集の方法については、「[Process Server の管理および監視](#)」を参照してください。

3.2.10. 追加の管理 Process Server のテンプレートデプロイメントの開始

OpenShift Web UI またはコマンドラインに必要なすべてのパラメーターを設定した後に、テンプレートのデプロイを実行します。

手順

使用している方法に応じて、以下の手順を実行します。

- OpenShift Web UI の場合は **Create** をクリックします。
 - **This will create resources that may have security or project behavior implications** メッセージが表示された場合は、**Create Anyway** をクリックします。
- コマンドラインに入力して、Enter キーを押します。

3.3. (任意) 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。たとえば、以下のようになります。

```
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。以下のデプロイメント設定は、この環境で影響を受ける可能性があります。

myapp はアプリケーション名に置き換えます。複数の Process Server デプロイメントが異なるアプリケーション名で存在する場合があります。

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。

第4章 RED HAT PROCESS AUTOMATION MANAGER ロールおよびユーザー

Business Central または Process Server にアクセスするには、サーバーを起動する前にユーザーを作成して適切なロールを割り当てます。

Business Central および Process Server は、JAVA 認証承認サービス(JAAS)ログインモジュールを使用してユーザーを認証します。Business Central と Process Server の両方が単一のインスタンスで実行されている場合は、同じ JAAS サブジェクトとセキュリティドメインを共有します。したがって、Business Central に対して認証されたユーザーは、Process Server にもアクセスできます。

ただし、Business Central と Process Server が異なるインスタンスで実行されている場合、JAAS ログインモジュールは両方に対して個別にトリガーされます。したがって、Business Central に対して認証されたユーザーは、Process Server にアクセスするために個別に認証する必要があります（例：Business Central でプロセス定義を表示または管理する場合）。ユーザーが Process Server で認証されていない場合は、ログファイルに 401 エラーが記録され、Business Central に **Invalid credentials to load data from remote server.Contact your system administrator.** メッセージが表示されます。

このセクションでは、利用可能な Red Hat Process Automation Manager ユーザーロールについて説明します。



注記

admin、**analyst**、**developer**、**manager**、**process-admin**、**user**、および **rest-all** のロールは Business Central に予約されています。**kie-server** ロールは Process Server 用に予約されています。このため、Business Central または Process Server のいずれか、またはそれら両方がインストールされているかどうかによって、利用可能なロールは異なります。

- **admin:** **admin** ロールを持つユーザーは Business Central 管理者です。管理者は、ユーザーの管理や、リポジトリの作成、クローン作成、および管理ができます。アプリケーションで必要な変更をすべて利用できます。**admin** ロールを持つユーザーは、Red Hat Process Automation Manager の全領域にアクセスできます。
- **analyst:** **analyst** ロールを持つユーザーには、すべてのハイレベル機能へのアクセスがあります。これらは、プロジェクトのモデリングと実行を行うことができます。ただし、このユーザーは、**Design → Projects** ビューでスペースに貢献者を追加したり、スペースを削除したりできません。**analyst** ロールを持つユーザーは、管理者向けの **Deploy → Execution Servers** ビューにアクセスできません。ただし、これらのユーザーは、ライブラリーパースペクティブにアクセスするときに **Deploy** ボタンを使用できます。
- **developer:** **developer** ロールを持つユーザーは、ほぼすべての機能にアクセスができ、ルール、モデル、プロセスフロー、フォーム、およびダッシュボードを管理できます。アセットリポジトリを管理し、プロジェクトを作成、ビルド、およびデプロイでき、Red Hat JBoss Developer Studio を使用してプロセスを表示できます。**developer** ロールが割り当てられているユーザーには、新規リポジトリの作成やクローン作成などの、特定の管理機能は表示されません。
- **manager:** **manager** ロールを持つユーザーはレポートを表示できます。このユーザーは通常、ビジネスプロセス、そのパフォーマンス、ビジネスインジケター、その他のビジネス関連のレポートに関する統計に関心があります。このルールを持つユーザーがアクセスできるのはプロセスおよびタスクのレポートに限られます。

- **process-admin: process-admin** ロールを持つユーザーは、ビジネスプロセス管理者です。ビジネスプロセス、ビジネスタスク、および実行エラーへの完全アクセスがあります。このユーザーは、ビジネスレポートを表示でき、タスク受信箱リストにアクセスできます。
- **user: user** ロールを持つユーザーは、タスクの受信箱リストで有効です。これには、現在実行しているプロセスの一部であるビジネスタスクも含まれます。このルールを持つユーザーはプロセスとタスクのレポートを確認して、プロセスを管理できます。
- **rest-all: rest-all** ロールを持つユーザーは、Business Central REST 機能にアクセスできます。
- **kie-server: kie-server** ロールを持つユーザーは Process Server (KIE サーバー) REST 機能へのアクセスがあります。このルールは、Business Central で Manage ビューおよび Track ビューにアクセスするユーザーに必要になります。

第5章 OPENSIFT テンプレートの参考資料

Red Hat Process Automation Manager は以下の OpenShift テンプレートを提供します。テンプレートにアクセスするには、Red Hat カスタマーポータル[の Software Downloads ページ](#)から、**rhpmam-7.4.0-openshift-templates.zip** の製品配信可能ファイルをダウンロードし、これを展開します。

- **rhpmam74-managed.yaml** は、高可用性 Business Central Monitoring インスタンス、Process Server、Process Server が使用する PostgreSQL インスタンスを提供します。**OpenShiftStartupStrategy** が有効になっているため、これらのインスタンスでも OpenShiftStartupStrategy が有効になっている限り、Business Central Monitoring インスタンスは同じプロジェクト内の他の Process Server インスタンスに自動的に接続できます。

この環境で使用される他のテンプレートに関する参考情報は、『Red Hat [OpenShift Container Platform への Red Hat Process Automation Manager イミュータブルサーバー環境のデプロイ](#)』の[参照セクション](#)を参照してください。

5.1. RHPAM74-MANAGED.YAML テンプレート

Red Hat Process Automation Manager 7.4 向けの、管理 HA 実稼働ランタイム環境のアプリケーションテンプレート

5.1.1. パラメーター

テンプレートを使用すると値を引き継ぐパラメーターを定義でき、パラメーターの参照時には、この値が代入されます。この値は、パラメーターの参照時には、この値が代入されます。参照はオブジェクト一覧フィールドの任意のテキストフィールドで定義できます。詳細は、[Openshift ドキュメント](#)を参照してください。

変数名	イメージの環境変数	説明	値の例	必須
APPLICATION_NAME	–	アプリケーションの名前。	myapp	True
MAVEN_MIRROR_URL	MAVEN_MIRROR_URL	KIE Server が使用する必要のある Maven ミラー。ミラーを設定する場合、このミラーにはサービスのビルドおよびデプロイに必要なすべてのアーティファクトを含める必要があります。	–	False
MAVEN_MIRROR_OF	MAVEN_MIRROR_OF	KIE Server の Maven ミラー設定。	external:*	False

変数名	イメージの環境変数	説明	値の例	必須
MAVEN_REPO_ID	MAVEN_REPO_ID	Maven リポジトリに使用する ID。これが設定されている場合は、MAVEN_MIRROR_OF に追加して、必要に応じて設定したミラーから除外できます。たとえば、external:*,!repo-rhpamcentr,!repo-custom などがあります。MAVEN_MIRROR_URL に設定されていても MAVEN_MIRROR_ID が設定されていない場合は、ID が無作為に生成され、MAVEN_MIRROR_OF では使用できません。	repo-custom	False
MAVEN_REPO_URL	MAVEN_REPO_URL	Maven リポジトリまたはサービスへの完全修飾 URL。	http://nexus.nexus-project.svc.cluster.local:8081/nexus/content/groups/public/	True
MAVEN_REPO_USERNAME	MAVEN_REPO_USERNAME	Maven リポジトリにアクセスするユーザー名 (必要な場合)	–	False
MAVEN_REPO_PASSWORD	MAVEN_REPO_PASSWORD	Maven リポジトリにアクセスするパスワード (必要な場合)。	–	False

変数名	イメージの環境変数	説明	値の例	必須
BUSINESS_CENTRAL_SERVICE	RHPAMCENTRAL_MAVEN_REPO_SERVICE	maven リポジトリーの使用状況など、サービスのルックアップができるように任意の Business Central に到達可能な場合に使用するサービス名 (必要な場合)	myapp-rhpamcentrmon	False
BUSINESS_CENTRAL_MAVEN_USERNAME	RHPAMCENTRAL_MAVEN_REPO_USERNAME	EAP 内の Business Central がホストする Maven サービスにアクセスするためのユーザー名	mavenUser	False
BUSINESS_CENTRAL_MAVEN_PASSWORD	RHPAMCENTRAL_MAVEN_REPO_PASSWORD	EAP 内の Business Central がホストする Maven サービスにアクセスするためのパスワード	maven!!	False
KIE_ADMIN_USER	KIE_ADMIN_USER	KIE 管理者のユーザー名	adminUser	False
KIE_ADMIN_PASSWORD	KIE_ADMIN_PASSWORD	KIE 管理者のパスワード	–	False
KIE_SERVER_USER	KIE_SERVER_USER	KIE サーバーのユーザー名 (org.kie.server.user システムプロパティを設定)	executionUser	False
KIE_SERVER_PASSWORD	KIE_SERVER_PASSWORD	KIE サーバーのパスワード。 (org.kie.server.password システムプロパティを設定)	–	False

変数名	イメージの環境変数	説明	値の例	必須
KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED	KIE_SERVER_CONTROLLER_OPENSHIFT_GLOBAL_DISCOVERY_ENABLED	true に設定すると、KIE Server のグローバル検出機能はオンになります (org.kie.server.controller.openshift.global.discovery.enabled システムプロパティを設定)。	false	False
KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	Business Central の OpenShift 統合がオンの場合は、このパラメーターを true に設定すると、OpenShift 内部サービスエンドポイント経由での KIE Server への接続が有効になります。 (org.kie.server.controller.openshift.prefer.kieserver.service システムプロパティを設定します)	true	False
KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL	KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL	KIE ServerTemplate Cache TTL (ミリ秒単位)。 (org.kie.server.controller.template.cache.ttl システムプロパティを設定)	60000	False

変数名	イメージの環境変数	説明	値の例	必須
IMAGE_STREAM_NAMESPACE	–	Red Hat ミドルウェアイメージの ImageStreams がインストールされている namespace これらの ImageStreams は通常 OpenShift の名前空間にインストールされています。ImageStreams を別の namespace/プロジェクトにインストールしている場合には、これを変更するだけで結構です。	openshift	True
KIE_SERVER_IMAGE_STREAM_NAME	–	KIE Server に使用するイメージストリームの名前。デフォルトは rhpam74-kieserver-openshift です。	rhpam74-kieserver-openshift	True
IMAGE_STREAM_TAG	–	イメージストリーム内のイメージへの名前付きポインター。デフォルトは 1.0 です。	1.0	True
KIE_SERVER_CONTROLLER_USER	KIE_SERVER_CONTROLLER_USER	KIE サーバーコントローラーのユーザー名。 (org.kie.server.controller.user システムプロパティーを設定)	controllerUser	False
KIE_SERVER_CONTROLLER_PASSWORD	KIE_SERVER_CONTROLLER_PASSWORD	KIE サーバーコントローラーのパスワード。 (org.kie.server.controller.pwd システムプロパティーを設定)	–	False

変数名	イメージの環境変数	説明	値の例	必須
KIE_SERVER_CONTROLLER_TOKEN	KIE_SERVER_CONTROLLER_TOKEN	ベアラー認証用の KIE Server コントローラートークン。 (org.kie.server.controller.token システムプロパティを設定)	–	False
KIE_SERVER_PERSISTENCE_DS	KIE_SERVER_PERSISTENCE_DS	KIE Server 永続データソース。 (org.kie.server.persistence.ds システムプロパティを設定)	java:/jboss/datasources/rhpam	False
POSTGRESQL_IMAGE_STREAM_NAMESPACE	–	PostgreSQL イメージの ImageStream がインストールされている名前空間。ImageStream は openshift namespace にすでにインストールされています。ImageStream を異なる namespace/プロジェクトにインストールしている場合にのみこれを変更する必要があります。デフォルトは openshift です。	openshift	False
POSTGRESQL_IMAGE_STREAM_TAG	–	PostgreSQL イメージのバージョン。これは PostgreSQL バージョンに対応するように意図されています。デフォルトは 10 です。	10	False
KIE_SERVER_POSTGRESQL_USER	RHPAM_USER_NAME	KIE サーバー PostgreSQL データベースのユーザー名。	rhpam	False

変数名	イメージの環境変数	説明	値の例	必須
KIE_SERVER_POSTGRESQL_PASSWORD	RHPAM_PASSWORD	KIE Server PostgreSQL データベースのパスワード。	–	False
KIE_SERVER_POSTGRESQL_DATABASE	RHPAM_DATABASE	KIE Server PostgreSQL データベース名。	rhpm7	False
POSTGRESQL_MAX_PREPARED_TRANSACTIONS	POSTGRESQL_MAX_PREPARED_TRANSACTIONS	PostgreSQL による XA トランザクションの処理を許可します。	100	True
DB_VOLUME_CAPACITY	–	データベースボリュームの永続ストレージのサイズ。	1Gi	True
KIE_SERVER_POSTGRESQL_DIALECT	KIE_SERVER_PERSISTENCE_DIALECT	KIE Server PostgreSQL Hibernate 方言。	org.hibernate.dialect.PostgreSQLDialect	True
KIE_SERVER_MODE	KIE_SERVER_MODE	KIE Server モード。有効な値は 'DEVELOPMENT' または 'PRODUCTION' です。実稼働モードでは、SNAPSHOT バージョンのアーティファクトは KIE Server にデプロイできず、既存のコンテナでアーティファクトのバージョンを変更することはできません。 (org.kie.server.mode システムプロパティを設定)	PRODUCTION	False

変数名	イメージの環境変数	説明	値の例	必須
KIE_MBEANS	KIE_MBEANS	KIE Server の mbeans の有効化/無効化 (システムプロパティー kie.mbeans および kie.scanner.mbeans を設定)。	enabled	False
DROOLS_SERVER_FILTER_CLASSES	DROOLS_SERVER_FILTER_CLASSES	KIE Server クラスのフィルターリング。 (org.drools.server.filter.classes システムプロパティーを設定)	true	False
PROMETHEUS_SERVER_EXT_DISABLED	PROMETHEUS_SERVER_EXT_DISABLED	false に設定すると、prometheus サーバー拡張が有効になります。 (org.kie.prometheus.server.ext.disabled システムプロパティーを設定)	false	False
BUSINESS_CENTRAL_HOSTNAME_HTTP	HOSTNAME_HTTP	http サービスルートのカスタムホスト名。デフォルトのホスト名を使用する場合は空白にします (例: insecure- <application-name>- rhpamcentrmon- <project>.<default-domain-suffix>)。	—	False

変数名	イメージの環境変数	説明	値の例	必須
BUSINESS_CENTRAL_HOSTNAME_HTTPS	HOSTNAME_HTTPS	https サービスルートのカスタムのホスト名。デフォルトのホスト名を使用する場合は空白にします (例: <application-name>-rhpamcentrmon-<project>.<default-domain-suffix>)。	—	False
KIE_SERVER_HOSTNAME_HTTP	HOSTNAME_HTTP	http サービスルートのカスタムホスト名。デフォルトホスト名の場合は空白にします (例: insecure-<application-name>-kieserver-<project>.<default-domain-suffix>)。	—	False
KIE_SERVER_HOSTNAME_HTTPS	HOSTNAME_HTTPS	https サービスルートのカスタムのホスト名。デフォルトホスト名の場合は空白にします (例: <application-name>-kieserver-<project>.<default-domain-suffix>)。	—	False
BUSINESS_CENTRAL_HTTPS_SECRET	—	Business Central のキーストアファイルが含まれるシークレットの名前。	businesscentral-app-secret	True
BUSINESS_CENTRAL_HTTPS_KEYSTORE	HTTPS_KEYSTORE	シークレット内のキーストアファイルの名前。	keystore.jks	False
BUSINESS_CENTRAL_HTTPS_NAME	HTTPS_NAME	サーバー証明書に関連付けられている名前	jboss	False

変数名	イメージの環境変数	説明	値の例	必須
BUSINESS_CENTRAL_HTTPS_PASSWORD	HTTPS_PASSWORD	キーストアおよび証明書のパスワード。	mykeystorepass	False
KIE_SERVER_HTTPS_SECRET	–	KIE Server のキーストアファイルが含まれるシークレットの名前。	kieserver-app-secret	True
KIE_SERVER_HTTPS_KEYSTORE	HTTPS_KEYSTORE	シークレット内のキーストアファイルの名前。	keystore.jks	False
KIE_SERVER_HTTPS_NAME	HTTPS_NAME	サーバー証明書に関連付けられている名前	jboss	False
KIE_SERVER_HTTPS_PASSWORD	HTTPS_PASSWORD	キーストアおよび証明書のパスワード。	mykeystorepass	False
KIE_SERVER_BYPASS_AUTH_USER	KIE_SERVER_BYPASS_AUTH_USER	KIE サーバーはタスク関連の操作は認証済みユーザーをスキップできる。 (org.kie.server.bypass.auth.user システムプロパティを設定)	false	False
TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL	TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL	EJB タイマーサービスのデータベースデータストアの更新間隔を設定します。	30000	False
BUSINESS_CENTRAL_MEMORY_LIMIT	–	Business Central Monitoring コンテナのメモリー制限	2Gi	False
KIE_SERVER_MEMORY_LIMIT	–	KIE Server のコンテナのメモリー制限。	1Gi	False

変数名	イメージの環境変数	説明	値の例	必須
BUSINESS_CENTRAL_MONITORING_CONTAINER_REPLICAS	–	Business Central Monitoring Container Replicas は、起動する Business Central Monitoring コンテナ数を定義します。	3	True
KIE_SERVER_CONTAINER_REPLICAS	–	KIE Server Container Replicas は、起動する KIE Server コンテナ数を定義します。	3	True
SSO_URL	SSO_URL	RH-SSO URL。	https://rh-sso.example.com/auth	False
SSO_REALM	SSO_REALM	RH-SSO レルム名。	–	False
BUSINESS_CENTRAL_SSO_CLIENT	SSO_CLIENT	Business Central Monitoring RH-SSO クライアント名。	–	False
BUSINESS_CENTRAL_SSO_SECRET	SSO_SECRET	Business Central Monitoring RH-SSO クライアントシークレット。	252793ed-7118-4ca8-8dab-5622fa97d892	False
KIE_SERVER_SSO_CLIENT	SSO_CLIENT	KIE Server の RH-SSO クライアント名。	–	False
KIE_SERVER_SSO_SECRET	SSO_SECRET	KIE Server の RH-SSO クライアントシークレット。	252793ed-7118-4ca8-8dab-5622fa97d892	False
SSO_USERNAME	SSO_USERNAME	クライアント作成に使用する RH-SSO レルムの管理者のユーザー名 (存在しない場合)	–	False

変数名	イメージの環境変数	説明	値の例	必須
SSO_PASSWORD	SSO_PASSWORD	クライアント作成に使用する RH-SSO レルムの管理者のパスワード。	–	False
SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO が無効な SSL 証明書の検証。	false	False
SSO_PRINCIPAL_ATTRIBUTE	SSO_PRINCIPAL_ATTRIBUTE	ユーザー名として使用する RH-SSO プリンシパル属性	preferred_username	False
AUTH_LDAP_URL	AUTH_LDAP_URL	認証用に接続する LDAP エンドポイント。	ldap://myldap.example.com	False
AUTH_LDAP_BIND_DN	AUTH_LDAP_BIND_DN	認証に使用するバインド DN	uid=admin,ou=users,ou=example,ou=com	False
AUTH_LDAP_BIND_CREDENTIAL	AUTH_LDAP_BIND_CREDENTIAL	認証に使用する LDAP の認証情報	パスワード	False
AUTH_LDAP_JAAS_SECURITY_DOMAIN	AUTH_LDAP_JAAS_SECURITY_DOMAIN	パスワードの復号に使用する JaasSecurityDomain の JMX ObjectName。	–	False
AUTH_LDAP_BASE_CTX_DN	AUTH_LDAP_BASE_CTX_DN	ユーザー検索を開始する最上位コンテキストの LDAP ベース DN	ou=users,ou=example,ou=com	False

変数名	イメージの環境変数	説明	値の例	必須
AUTH_LDAP_B ASE_FILTER	AUTH_LDAP_B ASE_FILTER	認証するユーザーのコンテキストの検索に使用するLDAP 検索フィルター。{0} 式を使用しているフィルターに、入力ユーザー名、またはログインモジュールコールバックから取得した userDN が置換されます。検索フィルターの一般的な例は (uid={0}) です。	(uid={0})	False
AUTH_LDAP_S EARCH_SCOPE	AUTH_LDAP_S EARCH_SCOPE	使用する検索範囲。	SUBTREE_SCO PE	False
AUTH_LDAP_S EARCH_TIME_L IMIT	AUTH_LDAP_S EARCH_TIME_L IMIT	ユーザーまたはロールの検索のタイムアウト (ミリ秒単位)。	10000	False
AUTH_LDAP_DI STINGUISHED_ NAME_ATTRIB UTE	AUTH_LDAP_DI STINGUISHED_ NAME_ATTRIB UTE	ユーザーの DN を含むユーザーエントリーの属性の名前。これは、ユーザー自身の DN に正しいユーザーマッピングを妨げる特殊文字 (バックスラッシュなど) が含まれる場合に必要になることがあります。属性が存在しない場合は、エントリーの DN が使用されます。	distinguishedName	False

変数名	イメージの環境変数	説明	値の例	必須
AUTH_LDAP_PARSE_USERNAME	AUTH_LDAP_PARSE_USERNAME	DN がユーザー名に対して解析されるかどうかを示すフラグ。true に設定した場合には、DN はユーザー名に対して解析されます。false に設定した場合には、DN はユーザー名に対して解析されません。このオプションは、usernameBeginString および usernameEndString とともに使用されます。	true	False
AUTH_LDAP_USERNAME_BEGIN_STRING	AUTH_LDAP_USERNAME_BEGIN_STRING	ユーザー名を公開するため、DN の最初から削除される文字列を定義します。このオプションは usernameEndString と合わせて使用し、parseUsername が true に設定されている場合にのみ考慮されます。	–	False
AUTH_LDAP_USERNAME_END_STRING	AUTH_LDAP_USERNAME_END_STRING	ユーザー名を公開するため、DN の最後から削除される文字列を定義します。このオプションは usernameEndString と合わせて使用し、parseUsername が true に設定されている場合にのみ考慮されます。	–	False
AUTH_LDAP_ROLE_ATTRIBUTE_ID	AUTH_LDAP_ROLE_ATTRIBUTE_ID	ユーザーロールを含む属性の名前。	memberOf	False

変数名	イメージの環境変数	説明	値の例	必須
AUTH_LDAP_ROLE_CTX_DN	AUTH_LDAP_ROLE_CTX_DN	ユーザーロールを検索するコンテキストの固定 DN。これは、実際のロールが存在する DN ではなく、ユーザーロールを含むオブジェクトが存在する DN です。たとえば、Microsoft Active Directory サーバーでは、これは、ユーザーアカウントが存在する DN です。	ou=groups,ou=example,ou=com	False
AUTH_LDAP_ROLE_FILTER	AUTH_LDAP_ROLE_FILTER	認証済みユーザーと関連付けられたロールを検索するために使用される検索フィルター。 {0} 式を使用しているフィルターに、入力ユーザー名、またはログインモジュールコールバックから取得した userDN が置換されます。認証済み userDN は {1} が使用されたフィルターに置き換えられます。入力ユーザー名に一致する検索フィルター例は (member={0}) です。認証済み userDN に一致する他の例は (member={1}) です。	(memberOf={1})	False

変数名	イメージの環境変数	説明	値の例	必須
AUTH_LDAP_ROLE_RECURSION	AUTH_LDAP_ROLE_RECURSION	ロール検索が一致するコンテキストで行われる再帰のレベル数。再帰を無効にするには、これを 0 に設定します。	1	False
AUTH_LDAP_DEFAULT_ROLE	AUTH_LDAP_DEFAULT_ROLE	認証された全ユーザーに対して含まれるロール	user	False
AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	ロール名を含む roleCtxDN コンテキスト内の属性の名前。 roleAttributesDN プロパティを true に設定すると、このプロパティはロールオブジェクトの名前属性の検索に使用されます。	name	False
AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	クエリーによって返された DN に roleNameAttributeID が含まれるかどうかを示すフラグ。true に設定した場合には、DN は roleNameAttributeID に対してチェックされます。false に設定すると、DN は roleNameAttributeID に対して確認されません。このフラグは LDAP クエリーのパフォーマンスを向上できます。	false	False

変数名	イメージの環境変数	説明	値の例	必須
AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	roleAttributeID に ロールオブジェクトの完全修飾 DN が含まれるかどうか。false の場合 は、コンテキスト名の roleNameAttribute Id 属性の値からこの ロール名が取得 されます。 Microsoft Active Directory などの特 定のディレクト リースキーマで は、この属性を true に設定する必 要があります。	false	False
AUTH_LDAP_REFERRAL_USE_R_ATTRIBUTE_ID_TO_CHECK	AUTH_LDAP_REFERRAL_USE_R_ATTRIBUTE_ID_TO_CHECK	リファーラル (referral) を使用し ない場合はこのオ プションを使用す る必要はありませ ん。リファーラル を使用し、ロール オブジェクトがリ ファーラル内部に あると、このオプ ションは特定の ロール (例: member) に対して 定義されたユー ザーが含まれる属 性名を示します。 ユーザーはこの属 性名の内容に対し て確認されます。 このオプションが 設定されていない とチェックは常に 失敗するため、 ロールオブジェク トはリファーラル ツリーに保存でき ません。	—	False

変数名	イメージの環境変数	説明	値の例	必須
AUTH_ROLE_MAPPER_ROLES_PROPERTIES	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	このパラメーターがある場合には、RoleMapping のログインモジュールで、指定したファイルを使用するように設定します。このパラメーターは、ロールを置換ロールに対してマップするプロパティファイルまたはリソースの完全修飾ファイルパスまたはファイル名を定義します。形式は original_role=role1,role2,role3 になります。	–	False
AUTH_ROLE_MAPPER_REPLACE_ROLE	AUTH_ROLE_MAPPER_REPLACE_ROLE	現在のロールを追加するか、マップされたロールに現在のロールを置き換えるか。true に設定した場合は、置き換えられます。	–	False

5.1.2. オブジェクト

CLI はさまざまなオブジェクトタイプをサポートします。これらのオブジェクトタイプの一覧や略語については、[Openshift ドキュメント](#) を参照してください。

5.1.2.1. サービス

サービスは、Pod の論理セットや、Pod にアクセスするためのポリシーを定義する抽象概念です。詳細は、[コンテナエンジンのドキュメント](#) を参照してください。

サービス	ポート	名前	説明
`\${APPLICATION_NAME}-rhpamcentrmon`	8080	http	すべての Business Central Monitoring Web サーバーのポート。
	8443	https	

サービス	ポート	名前	説明
\${APPLICATION_NAME}-rhpmcentrmon-ping	8888	ping	クラスターリング向けの JGroups ping ポート。
\${APPLICATION_NAME}-kieserver	8080	http	すべての KIE Server Web サーバーのポート。(1つ目の KIE server)
	8443	https	
\${APPLICATION_NAME}-kieserver-ping	8888	ping	クラスターリング向けの JGroups ping ポート。
\${APPLICATION_NAME}-postgresql	5432	–	最初のデータベースサーバーのポート。

5.1.2.2. ルート

ルートとは、**www.example.com** など、外部から到達可能なホスト名を指定して、サービスを公開する手段です。ルーターは、定義したルートや、サービスで特定したエンドポイントを使用して、外部のクライアントからアプリケーションに名前付きの接続を提供します。各ルートは、ルート名、サービスセレクター、セキュリティ設定 (任意) で設定されます。詳細は、[Openshift ドキュメント](#) を参照してください。

サービス	セキュリティ	ホスト名
insecure- \${APPLICATION_NAME}-rhpmcentrmon-http	なし	\${BUSINESS_CENTRAL_HOSTNAME_HTTP}
\${APPLICATION_NAME}-rhpmcentrmon-https	TLS パススルー	\${BUSINESS_CENTRAL_HOSTNAME_HTTPS}
insecure- \${APPLICATION_NAME}-kieserver-http	なし	\${KIE_SERVER_HOSTNAME_HTTP}
\${APPLICATION_NAME}-kieserver-https	TLS パススルー	\${KIE_SERVER_HOSTNAME_HTTPS}

5.1.2.3. デプロイメント設定

OpenShift のデプロイメントは、デプロイメント設定と呼ばれるユーザー定義のテンプレートをもとにするレプリケーションコントローラーです。デプロイメントは手動で作成されるか、トリガーされたイベントに対応するために作成されます。詳細は、[Openshift ドキュメント](#) を参照してください。

5.1.2.3.1. トリガー

トリガーは、OpenShift 内外を問わず、イベントが発生すると新規デプロイメントを作成するように促します。詳細は、[OpenShift ドキュメント](#) を参照してください。

Deployment	トリガー
<code>\${APPLICATION_NAME}-rhpamcentrmon</code>	ImageChange
<code>\${APPLICATION_NAME}-kieserver</code>	ImageChange
<code>\${APPLICATION_NAME}-postgresql</code>	ImageChange

5.1.2.3.2. レプリカ

レプリケーションコントローラーを使用すると、指定した数だけ、Pod のレプリカを一度に実行させることができます。レプリカが増えると、レプリケーションコントローラーが Pod の一部を終了させます。レプリカが足りない場合には、起動させます。詳細は、[コンテナエンジンのドキュメント](#) を参照してください。

Deployment	レプリカ
<code>\${APPLICATION_NAME}-rhpamcentrmon</code>	3
<code>\${APPLICATION_NAME}-kieserver</code>	3
<code>\${APPLICATION_NAME}-postgresql</code>	1

5.1.2.3.3. Pod テンプレート

5.1.2.3.3.1. サービスアカウント

サービスアカウントは、各プロジェクト内に存在する API オブジェクトです。他の API オブジェクトのように作成し、削除できます。詳細は、[OpenShift ドキュメント](#) を参照してください。

Deployment	サービスアカウント
<code>\${APPLICATION_NAME}-rhpamcentrmon</code>	<code>\${APPLICATION_NAME}-rhpamsvc</code>
<code>\${APPLICATION_NAME}-kieserver</code>	<code>\${APPLICATION_NAME}-rhpamsvc</code>

5.1.2.3.3.2. イメージ

デプロイメント	イメージ
<code>\${APPLICATION_NAME}-rhpamcentrmon</code>	<code>rhpam74-businesscentral-monitoring-openshift</code>
<code>\${APPLICATION_NAME}-kieserver</code>	<code>\${KIE_SERVER_IMAGE_STREAM_NAME}</code>

デプロイメント	イメージ
<code>\${APPLICATION_NAME}-postgresql</code>	postgresql

5.1.2.3.3.3. Readiness Probe

`${APPLICATION_NAME}-rhpamcentrmon`

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

`${APPLICATION_NAME}-kieserver`

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

`${APPLICATION_NAME}-postgresql`

`/usr/libexec/check-container`

5.1.2.3.3.4. Liveness Probe

`${APPLICATION_NAME}-rhpamcentrmon`

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

`${APPLICATION_NAME}-kieserver`

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

`${APPLICATION_NAME}-postgresql`

`/usr/libexec/check-container --live`

5.1.2.3.3.5. 公開されたポート

デプロイメント	名前	ポート	プロトコル
<code>\${APPLICATION_NAME}-rhpamcentrmon</code>	jolokia	8778	TCP
	http	8080	TCP
	https	8443	TCP
	ping	8888	TCP
<code>\${APPLICATION_NAME}-kieserver</code>	jolokia	8778	TCP

デプロイメント	名前	ポート	プロトコル
	http	8080	TCP
	https	8443	TCP
	ping	8888	TCP
\${APPLICATION_NAME}-postgresql	–	5432	TCP

5.1.2.3.3.6. イメージの環境変数

デプロイメント	変数名	説明	値の例
\${APPLICATION_NAME}-rhpamcentrmon	KIE_ADMIN_PWD	KIE 管理者のパスワード	\${KIE_ADMIN_PWD}
	KIE_ADMIN_USER	KIE 管理者のユーザー名	\${KIE_ADMIN_USER}
	KIE_SERVER_PWD	KIE サーバーのパスワード。 (org.kie.server.pwd システムプロパティーを設定)	\${KIE_SERVER_PWD}
	KIE_SERVER_USER	KIE サーバーのユーザー名 (org.kie.server.user システムプロパティーを設定)	\${KIE_SERVER_USER}
	MAVEN_MIRROR_URL	KIE Server が使用する必要のある Maven ミラー。ミラーを設定する場合、このミラーにはサービスのビルドおよびデプロイに必要なすべてのアーティファクトを含める必要があります。	\${MAVEN_MIRROR_URL}

デプロイメント	変数名	説明	値の例
	MAVEN_REPO_ID	Maven リポジトリに使用する ID。これが設定されている場合は、MAVEN_MIRROR_OF に追加して、必要に応じて設定したミラーから除外できます。たとえば、external:*,!repo-rhpamcentr,!repo-custom などがあります。 MAVEN_MIRROR_URL に設定されていても MAVEN_MIRROR_ID が設定されていない場合は、ID が無作為に生成され、MAVEN_MIRROR_OF では使用できません。	\${MAVEN_REPO_ID}
	MAVEN_REPO_URL	Maven リポジトリまたはサービスへの完全修飾 URL。	\${MAVEN_REPO_URL}
	MAVEN_REPO_USE_USERNAME	Maven リポジトリにアクセスするユーザー名 (必要な場合)	\${MAVEN_REPO_USERNAME}
	MAVEN_REPO_PASSWORD	Maven リポジトリにアクセスするパスワード (必要な場合)。	\${MAVEN_REPO_PASSWORD}
	KIE_SERVER_CONTROLLER_OPENSIFT_GLOBAL_DISCOVERY_ENABLED	true に設定すると、KIE Server のグローバル検出機能はオンになります (org.kie.server.controller.opensift.global.discovery.enabled システムプロパティを設定)。	\${KIE_SERVER_CONTROLLER_OPENSIFT_GLOBAL_DISCOVERY_ENABLED}

デプロイメント	変数名	説明	値の例
	KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE	Business Central の OpenShift 統合がオンの場合は、このパラメーターを true に設定すると、OpenShift 内部サービスエンドポイント経由での KIE Server への接続が有効になります。 (org.kie.server.controller.openshift.prefer.kieserver.service システムプロパティーを設定します)	`\${KIE_SERVER_CONTROLLER_OPENSHIFT_PREFER_KIESERVER_SERVICE}`
	KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL	KIE ServerTemplate Cache TTL (ミリ秒単位)。 (org.kie.server.controller.template.cache.ttl システムプロパティーを設定)	`\${KIE_SERVER_CONTROLLER_TEMPLATE_CACHE_TTL}`
	KIE_WORKBENCH_CONTROLLER_OPENSHIFT_ENABLED	–	true
	KIE_SERVER_CONTROLLER_USER	KIE サーバーコントローラーのユーザー名。 (org.kie.server.controller.user システムプロパティーを設定)	`\${KIE_SERVER_CONTROLLER_USER}`
	KIE_SERVER_CONTROLLER_PWD	KIE サーバーコントローラーのパスワード。 (org.kie.server.controller.pwd システムプロパティーを設定)	`\${KIE_SERVER_CONTROLLER_PWD}`
	KIE_SERVER_CONTROLLER_TOKEN	ベアラー認証用の KIE Server コントローラー トークン。 (org.kie.server.controller.token システムプロパティーを設定)	`\${KIE_SERVER_CONTROLLER_TOKEN}`
	HTTPS_KEYSTORE_DIR	–	/etc/businesscentral-secret-volume

デプロイメント	変数名	説明	値の例
	HTTPS_KEYSTORE	シークレット内のキーストアファイルの名前。	`\${BUSINESS_CENTRAL_HTTPS_KEYSTORE}`
	HTTPS_NAME	サーバー証明書に関連付けられている名前。	`\${BUSINESS_CENTRAL_HTTPS_NAME}`
	HTTPS_PASSWORD	キーストアおよび証明書のパスワード。	`\${BUSINESS_CENTRAL_HTTPS_PASSWORD}`
	JGROUPS_PING_PROTOCOL	–	openshift.DNS_PING
	OPENSIFT_DNS_PING_SERVICE_NAME	–	`\${APPLICATION_NAME}`-rhpacentrmon-ping
	OPENSIFT_DNS_PING_SERVICE_PORT	–	8888
	SSO_URL	RH-SSO URL。	`\${SSO_URL}`
	SSO_OPENIDCONNECT_DEPLOYMENTS	–	ROOT.war
	SSO_REALM	RH-SSO レルム名。	`\${SSO_REALM}`
	SSO_SECRET	Business Central Monitoring RH-SSO クライアントシークレット。	`\${BUSINESS_CENTRAL_SSO_SECRET}`
	SSO_CLIENT	Business Central Monitoring RH-SSO クライアント名。	`\${BUSINESS_CENTRAL_SSO_CLIENT}`
	SSO_USERNAME	クライアント作成に使用する RH-SSO レルムの管理者のユーザー名 (存在しない場合)	`\${SSO_USERNAME}`

デプロイメント	変数名	説明	値の例
	SSO_PASSWORD	クライアント作成に使用する RH-SSO レルムの管理者のパスワード。	\${SSO_PASSWORD}
	SSO_DISABLE_SSL_CERTIFICATE_VALIDATION	RH-SSO が無効な SSL 証明書の検証。	\${SSO_DISABLE_SSL_CERTIFICATE_VALIDATION}
	SSO_PRINCIPAL_ATTRIBUTE	ユーザー名として使用する RH-SSO プリンシパル属性	\${SSO_PRINCIPAL_ATTRIBUTE}
	HOSTNAME_HTTP	http サービスルートのカスタムホスト名。デフォルトのホスト名を使用する場合は空白にします (例: insecure- <application-name>- rhpamcentrmon- <project>.<default-domain-suffix>)。	\${BUSINESS_CENTRAL_HOSTNAME_HTTP}
	HOSTNAME_HTTPS	https サービスルートのカスタムのホスト名。デフォルトのホスト名を使用する場合は空白にします (例: <application-name>- rhpamcentrmon- <project>.<default-domain-suffix>)。	\${BUSINESS_CENTRAL_HOSTNAME_HTTPS}
	AUTH_LDAP_URL	認証用に接続する LDAP エンドポイント。	\${AUTH_LDAP_URL}
	AUTH_LDAP_BIND_DN	認証に使用するバインド DN	\${AUTH_LDAP_BIND_DN}
	AUTH_LDAP_BIND_CREDENTIAL	認証に使用する LDAP の認証情報	\${AUTH_LDAP_BIND_CREDENTIAL}
	AUTH_LDAP_JAAS_SECURITY_DOMAIN	パスワードの復号に使用する JaasSecurityDomain の JMX ObjectName。	\${AUTH_LDAP_JAAS_SECURITY_DOMAIN}
	AUTH_LDAP_BASE_CTX_DN	ユーザー検索を開始する最上位コンテキストの LDAP ベース DN	\${AUTH_LDAP_BASE_CTX_DN}

デプロイメント	変数名	説明	値の例
	AUTH_LDAP_BASE_FILTER	認証するユーザーのコンテキストの検索に使用する LDAP 検索フィルター。{0} 式を使用しているフィルターに、入力ユーザー名、またはログインモジュールコールバックから取得した userDN が置換されます。検索フィルターの一般的な例は (uid={0}) です。	`\${AUTH_LDAP_BASE_FILTER}`
	AUTH_LDAP_SEARCH_SCOPE	使用する検索範囲。	`\${AUTH_LDAP_SEARCH_SCOPE}`
	AUTH_LDAP_SEARCH_TIME_LIMIT	ユーザーまたはロールの検索のタイムアウト (ミリ秒単位)。	`\${AUTH_LDAP_SEARCH_TIME_LIMIT}`
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	ユーザーの DN を含むユーザーエントリーの属性の名前。これは、ユーザー自身の DN に正しいユーザーマッピングを妨げる特殊文字 (バックスラッシュなど) が含まれる場合に必要になることがあります。属性が存在しない場合は、エントリーの DN が使用されます。	`\${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}`
	AUTH_LDAP_PARSE_USERNAME	DN がユーザー名に対して解析されるかどうかを示すフラグ。true に設定した場合には、DN はユーザー名に対して解析されます。false に設定した場合には、DN はユーザー名に対して解析されません。このオプションは、usernameBeginString および usernameEndString とともに使用されます。	`\${AUTH_LDAP_PARSE_USERNAME}`

デプロイメント	変数名	説明	値の例
	AUTH_LDAP_USER_NAME_BEGIN_STRING	ユーザー名を公開するため、DN の最初から削除される文字列を定義します。このオプションは usernameEndString と合わせて使用し、parseUsername が true に設定されている場合にのみ考慮されます。	`\${AUTH_LDAP_USER_NAME_BEGIN_STRING}`
	AUTH_LDAP_USER_NAME_END_STRING	ユーザー名を公開するため、DN の最後から削除される文字列を定義します。このオプションは usernameEndString と合わせて使用し、parseUsername が true に設定されている場合にのみ考慮されます。	`\${AUTH_LDAP_USER_NAME_END_STRING}`
	AUTH_LDAP_ROLE_ATTRIBUTE_ID	ユーザーロールを含む属性の名前。	`\${AUTH_LDAP_ROLE_ATTRIBUTE_ID}`
	AUTH_LDAP_ROLE_S_CTX_DN	ユーザーロールを検索するコンテキストの固定 DN。これは、実際のロールが存在する DN ではなく、ユーザーロールを含むオブジェクトが存在する DN です。たとえば、Microsoft Active Directory サーバーでは、これは、ユーザーアカウントが存在する DN です。	`\${AUTH_LDAP_ROLE_S_CTX_DN}`

デプロイメント	変数名	説明	値の例
	AUTH_LDAP_ROLE_FILTER	認証済みユーザーと関連付けられたロールを検索するために使用される検索フィルター。{0} 式を使用しているフィルターに、入力ユーザー名、またはログインモジュールコールバックから取得した userDN が置換されます。認証済み userDN は {1} が使用されたフィルターに置き換えられます。入力ユーザー名に一致する検索フィルター例は (member={0}) です。認証済み userDN に一致する他の例は (member={1}) です。	`\${AUTH_LDAP_ROLE_FILTER}`
	AUTH_LDAP_ROLE_RECURSION	ロール検索が一致するコンテキストで行われる再帰のレベル数。再帰を無効にするには、これを 0 に設定します。	`\${AUTH_LDAP_ROLE_RECURSION}`
	AUTH_LDAP_DEFAULT_ROLE	認証された全ユーザーに対して含まれるロール	`\${AUTH_LDAP_DEFAULT_ROLE}`
	AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	ロール名を含む roleCtxDN コンテキスト内の属性の名前。roleAttributesDN プロパティを true に設定すると、このプロパティはロールオブジェクトの名前属性の検索に使用されます。	`\${AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID}`

デプロイメント	変数名	説明	値の例
	AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	クエリーによって返された DN に roleNameAttributeID が含まれるかどうかを示すフラグ。true に設定した場合には、DN は roleNameAttributeID に対してチェックされます。false に設定すると、DN は roleNameAttributeID に対して確認されません。このフラグは LDAP クエリーのパフォーマンスを向上できます。	`\${AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN}`
	AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	roleAttributeID にロールオブジェクトの完全修飾 DN が含まれるかどうか。false の場合は、コンテキスト名の roleNameAttributeID 属性の値からこのロール名が取得されます。Microsoft Active Directory などの特定のディレクトリースキーマでは、この属性を true に設定する必要があります。	`\${AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN}`
	AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	リファール (referral) を使用しない場合はこのオプションを使用する必要はありません。リファールを使用し、ロールオブジェクトがリファール内部にあると、このオプションは特定のロール (例: member) に対して定義されたユーザーが含まれる属性名を示します。ユーザーはこの属性名の内容に対して確認されます。このオプションが設定されていないとチェックは常に失敗するため、ロールオブジェクトはリファールツリーに保存できません。	`\${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}`

デプロイメント	変数名	説明	値の例
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	このパラメーターがある場合には、RoleMappingのログインモジュールで、指定したファイルを使用するように設定します。このパラメーターは、ロールを置換ロールに対してマップするプロパティファイルまたはリソースの完全修飾ファイルパスまたはファイル名を定義します。形式は original_role=role1,role2,role3 になります。	`\${AUTH_ROLE_MAPPER_ROLES_PROPERTIES}`
	AUTH_ROLE_MAPPER_REPLACE_ROLE	現在のロールを追加するか、マップされたロールに現在のロールを置き換えるか。true に設定した場合は、置き換えられません。	`\${AUTH_ROLE_MAPPER_REPLACE_ROLE}`
`\${APPLICATION_NAME}`-kieserver	WORKBENCH_SERVICE_NAME	–	`\${APPLICATION_NAME}`-rhpamcentrmon
	KIE_ADMIN_USER	KIE 管理者のユーザー名	`\${KIE_ADMIN_USER}`
	KIE_ADMIN_PWD	KIE 管理者のパスワード	`\${KIE_ADMIN_PWD}`
	KIE_SERVER_MODE	KIE Server モード。有効な値は 'DEVELOPMENT' または 'PRODUCTION' です。実稼働モードでは、SNAPSHOT バージョンのアーティファクトは KIE Server にデプロイできず、既存のコンテナでアーティファクトのバージョンを変更することはできません。 (org.kie.server.mode システムプロパティを設定)	`\${KIE_SERVER_MODE}`

デプロイメント	変数名	説明	値の例
	KIE_MBEANS	KIE Server の mbeans の有効化/無効化 (システムプロパティー kie.mbeans および kie.scanner.mbeans を設定)。	`\${KIE_MBEANS}`
	DROOLS_SERVER_FILTER_CLASSES	KIE Server クラスのフィルタリング。 (org.drools.server.filter.classes システムプロパティーを設定)	`\${DROOLS_SERVER_FILTER_CLASSES}`
	PROMETHEUS_SERVER_EXT_DISABLED	false に設定すると、prometheus サーバー拡張が有効になります。 (org.kie.prometheus.server.ext.disabled システムプロパティーを設定)	`\${PROMETHEUS_SERVER_EXT_DISABLED}`
	KIE_SERVER_BYPASS_AUTH_USER	KIE サーバーはタスク関連の操作は認証済みユーザーをスキップできる。 (org.kie.server.bypass.auth.user システムプロパティーを設定)	`\${KIE_SERVER_BYPASS_AUTH_USER}`
	KIE_SERVER_ID	–	–
	KIE_SERVER_ROUTE_NAME	–	`\${APPLICATION_NAME}-kieserver`
	KIE_SERVER_STARTUP_STRATEGY	–	OpenShiftStartupStrategy
	KIE_SERVER_USER	KIE サーバーのユーザー名 (org.kie.server.user システムプロパティーを設定)	`\${KIE_SERVER_USER}`
	KIE_SERVER_PWD	KIE サーバーのパスワード。 (org.kie.server.pwd システムプロパティーを設定)	`\${KIE_SERVER_PWD}`

デプロイメント	変数名	説明	値の例
	MAVEN_MIRROR_URL	KIE Server が使用する必要のある Maven ミラー。ミラーを設定する場合、このミラーにはサービスのビルドおよびデプロイに必要なすべてのアーティファクトを含める必要があります。	`\${MAVEN_MIRROR_URL}`
	MAVEN_MIRROR_OFF	KIE Server の Maven ミラー設定。	`\${MAVEN_MIRROR_OFF}`
	MAVEN_REPOS	–	RHPAMCENTR,EXTERNAL
	RHPAMCENTR_MAVEN_REPO_ID	–	repo-rhpamcentr
	RHPAMCENTR_MAVEN_REPO_SERVICE	maven リポジトリの使用状況など、サービスのルックアップができるように任意の Business Central に到達可能な場合に使用するサービス名 (必要な場合)	`\${BUSINESS_CENTRAL_SERVICE}`
	RHPAMCENTR_MAVEN_REPO_PATH	–	/maven2/
	RHPAMCENTR_MAVEN_REPO_USERNAME	EAP 内の Business Central がホストする Maven サービスにアクセスするためのユーザー名	`\${BUSINESS_CENTRAL_MAVEN_USERNAME}`
	RHPAMCENTR_MAVEN_REPO_PASSWORD	EAP 内の Business Central がホストする Maven サービスにアクセスするためのパスワード	`\${BUSINESS_CENTRAL_MAVEN_PASSWORD}`

デプロイメント	変数名	説明	値の例
	EXTERNAL_MAVEN_REPO_ID	Maven リポジトリに使用する ID。これが設定されている場合は、MAVEN_MIRROR_OF に追加して、必要に応じて設定したミラーから除外できます。たとえば、external:*,!repo-rhpamcentr,!repo-custom などがあります。 MAVEN_MIRROR_URL に設定されていても MAVEN_MIRROR_ID が設定されていない場合は、ID が無作為に生成され、MAVEN_MIRROR_OF では使用できません。	\${MAVEN_REPO_ID}
	EXTERNAL_MAVEN_REPO_URL	Maven リポジトリまたはサービスへの完全修飾 URL。	\${MAVEN_REPO_URL}
	EXTERNAL_MAVEN_REPO_USERNAME	Maven リポジトリにアクセスするユーザー名 (必要な場合)	\${MAVEN_REPO_USERNAME}
	EXTERNAL_MAVEN_REPO_PASSWORD	Maven リポジトリにアクセスするパスワード (必要な場合)。	\${MAVEN_REPO_PASSWORD}
	KIE_SERVER_PERSISTENCE_DS	KIE Server 永続データソース。 (org.kie.server.persistence.ds システムプロパティを設定)	\${KIE_SERVER_PERSISTENCE_DS}
	DATASOURCES	–	RHPAM
	RHPAM_JNDI	KIE Server 永続データソース。 (org.kie.server.persistence.ds システムプロパティを設定)	\${KIE_SERVER_PERSISTENCE_DS}
	RHPAM_JTA	–	true

デプロイメント	変数名	説明	値の例
	RHPAM_DATABASE	KIE Server PostgreSQL データベース名。	\${KIE_SERVER_POSTGRESQL_DB}
	RHPAM_DRIVER	–	postgresql
	KIE_SERVER_PERSISTENCE_DIALECT	KIE Server PostgreSQL Hibernate 方言。	\${KIE_SERVER_POSTGRESQL_DIALECT}
	RHPAM_USERNAME	KIE サーバー PostgreSQL データベースのユーザー名。	\${KIE_SERVER_POSTGRESQL_USER}
	RHPAM_PASSWORD	KIE Server PostgreSQL データベースのパスワード。	\${KIE_SERVER_POSTGRESQL_PWD}
	RHPAM_SERVICE_HOST	–	\${APPLICATION_NAME}-postgresql
	RHPAM_SERVICE_PORT	–	5432
	TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL	EJB タイマーサービスのデータベースデータストアの更新間隔を設定します。	\${TIMER_SERVICE_DATA_STORE_REFRESH_INTERVAL}
	HTTPS_KEYSTORE_DIR	–	/etc/kieserver-secret-volume
	HTTPS_KEYSTORE	シークレット内のキーストアファイルの名前。	\${KIE_SERVER_HTTPS_KEYSTORE}
	HTTPS_NAME	サーバー証明書に関連付けられている名前	\${KIE_SERVER_HTTPS_NAME}
	HTTPS_PASSWORD	キーストアおよび証明書のパスワード。	\${KIE_SERVER_HTTPS_PASSWORD}
	JGROUPS_PING_PROTOCOL	–	openshift.DNS_PING
	OPENSIFT_DNS_PING_SERVICE_NAME	–	\${APPLICATION_NAME}-kieserver-ping

デプロイメント	変数名	説明	値の例
	OPENSIFT_DNS_PI NG_SERVICE_PORT	–	8888
	SSO_URL	RH-SSO URL。	\${SSO_URL}
	SSO_OPENIDCONN ECT_DEPLOYMENT S	–	ROOT.war
	SSO_REALM	RH-SSO レルム名。	\${SSO_REALM}
	SSO_SECRET	KIE Server の RH-SSO クライアントシークレッ ト。	\${KIE_SERVER_SSO _SECRET}
	SSO_CLIENT	KIE Server の RH-SSO クライアント名。	\${KIE_SERVER_SSO _CLIENT}
	SSO_USERNAME	クライアント作成に使用 する RH-SSO レルムの 管理者のユーザー名 (存 在しない場合)	\${SSO_USERNAME}
	SSO_PASSWORD	クライアント作成に使用 する RH-SSO レルムの 管理者のパスワード。	\${SSO_PASSWORD}
	SSO_DISABLE_SSL_ CERTIFICATE_VALI DATION	RH-SSO が無効な SSL 証明書の検証。	\${SSO_DISABLE_SS L_CERTIFICATE_VA LIDATION}
	SSO_PRINCIPAL_AT TRIBUTE	ユーザー名として使用す る RH-SSO プリンシパ ル属性	\${SSO_PRINCIPAL_ ATTRIBUTE}
	HOSTNAME_HTTP	http サービスルートの カスタムホスト名。デ フォルトホスト名の場合 は空白にします (例: insecure-<application- name>-kieserver- <project>.<default- domain-suffix>)。	\${KIE_SERVER_HOS TNAME_HTTP}

デプロイメント	変数名	説明	値の例
	HOSTNAME_HTTPS	https サービスルートのカスタムのホスト名。デフォルトホスト名の場合は空白にします (例: <application-name>-kieserver-<project>.<default-domain-suffix>。	`\${KIE_SERVER_HOSTNAME_HTTPS}`
	AUTH_LDAP_URL	認証用に接続する LDAP エンドポイント。	`\${AUTH_LDAP_URL}`
	AUTH_LDAP_BIND_DN	認証に使用するバインド DN	`\${AUTH_LDAP_BIND_DN}`
	AUTH_LDAP_BIND_CREDENTIAL	認証に使用する LDAP の認証情報	`\${AUTH_LDAP_BIND_CREDENTIAL}`
	AUTH_LDAP_JAAS_SECURITY_DOMAIN	パスワードの復号に使用する JaasSecurityDomain の JMX ObjectName。	`\${AUTH_LDAP_JAAS_SECURITY_DOMAIN}`
	AUTH_LDAP_BASE_CTX_DN	ユーザー検索を開始する最上位コンテキストの LDAP ベース DN	`\${AUTH_LDAP_BASE_CTX_DN}`
	AUTH_LDAP_BASE_FILTER	認証するユーザーのコンテキストの検索に使用する LDAP 検索フィルター。{0} 式を使用しているフィルターに、入力ユーザー名、またはログインモジュールコールバックから取得した userDN が置換されます。検索フィルターの一般的な例は (uid={0}) です。	`\${AUTH_LDAP_BASE_FILTER}`
	AUTH_LDAP_SEARCH_SCOPE	使用する検索範囲。	`\${AUTH_LDAP_SEARCH_SCOPE}`
	AUTH_LDAP_SEARCH_TIME_LIMIT	ユーザーまたはロールの検索のタイムアウト (ミリ秒単位)。	`\${AUTH_LDAP_SEARCH_TIME_LIMIT}`

デプロイメント	変数名	説明	値の例
	AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE	ユーザーの DN を含むユーザーエントリーの属性の名前。これは、ユーザー自身の DN に正しいユーザーマッピングを妨げる特殊文字 (バックスラッシュなど) が含まれる場合に必要になることがあります。属性が存在しない場合は、エントリーの DN が使用されます。	`\${AUTH_LDAP_DISTINGUISHED_NAME_ATTRIBUTE}`
	AUTH_LDAP_PARSE_USERNAME	DN がユーザー名に対して解析されるかどうかを示すフラグ。true に設定した場合には、DN はユーザー名に対して解析されます。false に設定した場合には、DN はユーザー名に対して解析されません。このオプションは、usernameBeginString および usernameEndString とともに使用されます。	`\${AUTH_LDAP_PARSE_USERNAME}`
	AUTH_LDAP_USERNAME_BEGIN_STRING	ユーザー名を公開するため、DN の最初から削除される文字列を定義します。このオプションは usernameEndString と合わせて使用し、parseUsername が true に設定されている場合のみ考慮されます。	`\${AUTH_LDAP_USERNAME_BEGIN_STRING}`
	AUTH_LDAP_USERNAME_END_STRING	ユーザー名を公開するため、DN の最後から削除される文字列を定義します。このオプションは usernameEndString と合わせて使用し、parseUsername が true に設定されている場合のみ考慮されます。	`\${AUTH_LDAP_USERNAME_END_STRING}`
	AUTH_LDAP_ROLE_ATTRIBUTE_ID	ユーザーロールを含む属性の名前。	`\${AUTH_LDAP_ROLE_ATTRIBUTE_ID}`

デプロイメント	変数名	説明	値の例
	AUTH_LDAP_ROLE_S_CTX_DN	ユーザーロールを検索するコンテキストの固定 DN。これは、実際のロールが存在する DN ではなく、ユーザーロールを含むオブジェクトが存在する DN です。たとえば、Microsoft Active Directory サーバーでは、これは、ユーザーアカウントが存在する DN です。	`\${AUTH_LDAP_ROLE_S_CTX_DN}`
	AUTH_LDAP_ROLE_FILTER	認証済みユーザーと関連付けられたロールを検索するために使用される検索フィルター。{0} 式を使用しているフィルターに、入力ユーザー名、またはログインモジュールコールバックから取得した userDN が置換されます。認証済み userDN は {1} が使用されたフィルターに置き換えられます。入力ユーザー名に一致する検索フィルター例は (member={0}) です。認証済み userDN に一致する他の例は (member={1}) です。	`\${AUTH_LDAP_ROLE_FILTER}`
	AUTH_LDAP_ROLE_RECURSION	ロール検索が一致するコンテキストで行われる再帰のレベル数。再帰を無効にするには、これを 0 に設定します。	`\${AUTH_LDAP_ROLE_RECURSION}`
	AUTH_LDAP_DEFAULT_ROLE	認証された全ユーザーに対して含まれるロール	`\${AUTH_LDAP_DEFAULT_ROLE}`
	AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID	ロール名を含む roleCtxDN コンテキスト内の属性の名前。roleAttributesDN プロパティーを true に設定すると、このプロパティーはロールオブジェクトの名前属性の検索に使用されます。	`\${AUTH_LDAP_ROLE_NAME_ATTRIBUTE_ID}`

デプロイメント	変数名	説明	値の例
	AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN	クエリーによって返された DN に roleNameAttributeID が含まれるかどうかを示すフラグ。true に設定した場合には、DN は roleNameAttributeID に対してチェックされます。false に設定すると、DN は roleNameAttributeID に対して確認されません。このフラグは LDAP クエリーのパフォーマンスを向上できます。	`\${AUTH_LDAP_PARSE_ROLE_NAME_FROM_DN}`
	AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN	roleAttributeID にロールオブジェクトの完全修飾 DN が含まれるかどうか。false の場合は、コンテキスト名の roleNameAttributeID 属性の値からこのロール名が取得されます。Microsoft Active Directory などの特定のディレクトリースキーマでは、この属性を true に設定する必要があります。	`\${AUTH_LDAP_ROLE_ATTRIBUTE_IS_DN}`
	AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK	リファール (referral) を使用しない場合はこのオプションを使用する必要はありません。リファールを使用し、ロールオブジェクトがリファール内部にあると、このオプションは特定のロール (例: member) に対して定義されたユーザーが含まれる属性名を示します。ユーザーはこの属性名の内容に対して確認されます。このオプションが設定されていないとチェックは常に失敗するため、ロールオブジェクトはリファールツリーに保存できません。	`\${AUTH_LDAP_REFERRAL_USER_ATTRIBUTE_ID_TO_CHECK}`

デプロイメント	変数名	説明	値の例
	AUTH_ROLE_MAPPER_ROLES_PROPERTIES	このパラメーターがある場合には、RoleMapping のログインモジュールで、指定したファイルを使用するように設定します。このパラメーターは、ロールを置換ロールに対してマップするプロパティファイルまたはリソースの完全修飾ファイルパスまたはファイル名を定義します。形式は original_role=role1,role2,role3 になります。	`\${AUTH_ROLE_MAPPER_ROLES_PROPERTIES}`
	AUTH_ROLE_MAPPER_REPLACE_ROLE	現在のロールを追加するか、マップされたロールに現在のロールを置き換えるか。true に設定した場合は、置き換えられません。	`\${AUTH_ROLE_MAPPER_REPLACE_ROLE}`
`\${APPLICATION_NAME}`-postgresql	POSTGRESQL_USER	KIE サーバー PostgreSQL データベースのユーザー名。	`\${KIE_SERVER_POSTGRESQL_USER}`
	POSTGRESQL_PASSWORD	KIE Server PostgreSQL データベースのパスワード。	`\${KIE_SERVER_POSTGRESQL_PWD}`
	POSTGRESQL_DATABASE	KIE Server PostgreSQL データベース名。	`\${KIE_SERVER_POSTGRESQL_DB}`
	POSTGRESQL_MAX_PREPARED_TRANSACTIONS	PostgreSQL による XA トランザクションの処理を許可します。	`\${POSTGRESQL_MAX_PREPARED_TRANSACTIONS}`

5.1.2.3.3.7. ボリューム

デプロイメント	名前	mountPath	目的	readOnly
`\${APPLICATION_NAME}`-rhpamcentrmon	businesscentral-keystore-volume	/etc/businesscentral-secret-volume	ssl certs	True
`\${APPLICATION_NAME}`-kieserver	kieserver-keystore-volume	/etc/kieserver-secret-volume	ssl certs	True

デプロイメント	名前	mountPath	目的	readOnly
<code>\${APPLICATION_NAME}-postgresql</code>	<code>\${APPLICATION_NAME}-postgresql-pvol</code>	<code>/var/lib/pgsql/data</code>	postgresql	false

5.1.2.4. 外部の依存関係

5.1.2.4.1. ボリューム要求

PersistentVolume オブジェクトは、OpenShift クラスターのストレージリソースです。管理者が GCE Persistent Disks、AWS Elastic Block Store (EBS)、NFS マウントなどのソースから **PersistentVolume** オブジェクトを作成して、ストレージをプロビジョニングします。詳細は、[OpenShift ドキュメント](#) を参照してください。

名前	アクセスモード
<code>\${APPLICATION_NAME}-postgresql-claim</code>	ReadWriteOnce
<code>\${APPLICATION_NAME}-rhpamcentr-claim</code>	ReadWriteMany

5.1.2.4.2. シークレット

このテンプレートでは、アプリケーションを実行するために以下のシークレットをインストールする必要があります。

```
businesscentral-app-secret kieserver-app-secret
```

5.2. OPENSIFT の使用に関するクイックリファレンス

Red Hat OpenShift Container Platform で Red Hat Process Automation Manager テンプレートをデプロイし、モニターし、管理し、デプロイ解除するには、OpenShift Web コンソールまたは **oc** コマンドを使用できます。

Web コンソールの使用に関する説明は、[Web コンソールを使用したイメージの作成およびビルド](#) を参照してください。

oc コマンドの使用方法に関する詳細は、[CLI リファレンス](#) を参照してください。次のコマンドが必要になる可能性があります。

- プロジェクトを作成するには、以下のコマンドを使用します。

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

詳細は、[CLI を使用したプロジェクトの作成](#) を参照してください。

- テンプレートをデプロイするには (またはテンプレートからアプリケーションを作成するには)、以下のコマンドを実行します。

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

詳細は、[CLI を使用したアプリケーションの作成](#) を参照してください。

- プロジェクト内のアクティブな Pod の一覧を表示するには、以下のコマンドを使用します。

```
$ oc get pods
```

- Pod のデプロイメントが完了し、実行中の状態になっているかどうかなど、Pod の現在のステータスを表示するには、以下のコマンドを使用します。

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

oc describe コマンドを使用して、他のオブジェクトの現在のステータスを表示できます。詳細は、[アプリケーションの変更操作](#) を参照してください。

- Pod のログを表示するには、以下のコマンドを使用します。

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

- デプロイメントログを表示するには、テンプレート参照で **DeploymentConfig** 名を検索し、以下のコマンドを入力します。

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

詳細は、[デプロイメントログの表示](#) を参照してください。

- ビルドログを表示するには、テンプレート参照で **BuildConfig** 名を検索し、以下のコマンドを入力します。

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

詳細は、[ビルドログのアクセス](#) を参照してください。

- アプリケーションの Pod をスケーリングするには、テンプレート参照で **DeploymentConfig** 名を検索し、以下のコマンドを入力します。

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

詳細は、[手動スケーリング](#) を参照してください。

- アプリケーションのデプロイメントを解除するには、以下のコマンドを使用してプロジェクトを削除します。

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

または、**oc delete** コマンドを使用して、Pod またはレプリケーションコントローラーなど、アプリケーションの一部を削除できます。詳細は、[アプリケーションの修正操作](#) を参照してください。

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）