



JBoss Enterprise BRMS Platform 5

Guide d'administration BRMS

Pour les administrateurs JBoss

Édition 5.2.0

JBoss Enterprise BRMS Platform 5 Guide d'administration BRMS

Pour les administrateurs JBoss
Édition 5.2.0

Landmann
rlandmann@redhat.com

Notice légale

Copyright © 2010 Red Hat, Inc..

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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, 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 Software Collections 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.

Résumé

Ce document fournit des instructions d'installation, de configuration et de personnalisation pour la plateforme JBoss Enterprise BRMS.

Table des matières

CHAPITRE 1. INTRODUCTION	3
CHAPITRE 2. INSTALLATION	4
CHAPITRE 3. LOCALISATION	17
CHAPITRE 4. CONFIGURATION DE LA BASE DE DONNÉES	19
CHAPITRE 5. GESTION DES DONNÉES	28
CHAPITRE 6. SÉCURITÉ	32
CHAPITRE 7. OUVERTURE DE SESSION	44
CHAPITRE 8. PERSONNALISATION	45
CHAPITRE 9. SURVEILLANCE	47
ANNEXE A. DÉPLOYER JBOSS BRMS SUR EWS	49
ANNEXE B. INSTALLER APACHE ANT	51
ANNEXE C. EXEMPLE DE CONFIGURATION DU GESTIONNAIRE DE PERSISTANCE	53
ANNEXE D. HISTORIQUE DE RÉVISION	55

CHAPITRE 1. INTRODUCTION

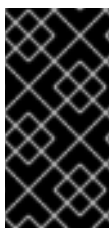
1.1. QUOI DE NEUF DANS CETTE ÉDITION ?

Tableau 1.1. Quoi de Neuf dans cette Édition ?

Fonctionnalités	Changements
Section 2.2.3, « Installer et activer ModeShape »	Une nouvelle section sur l'installation de Modeshape comme référentiel
Section 4.2, « Configuration de la Database de ModeShape »	Une nouvelle section sur la configuration d'une source de données pour Modeshape
Section 4.3.2, « Configurer BRMS pour qu'il utilise un RDBMS externe. »	Une nouvelle section sur la configuration de BRMS pour utiliser un RDBMS Externe
Chapitre 9, <i>Surveillance</i>	Nouveau chapitre sur la surveillance BRMS par un plug-in de réseau d'opérations JBoss
Annexe A, <i>Déployer JBoss BRMS sur EWS</i>	Un nouvelle section annexe avec des instructions pour installer JBoss BRMS sur JBoss Enterprise Web Server

1.2. JBOSS ENTERPRISE BRMS PLATFORM

La plate-forme JBoss Enterprise BRMS est un *Business Rules Management System* basé- règles de JBoss. Il offre une solution côté serveur, multi-utilisateur pour la gestion, le stockage, l'édition et le déploiement de règles et d'autres actifs de JBoss. Une interface conviviale web est fournie, ainsi que l'intégration pour JBoss Developer Studio 4.



IMPORTANT

À plusieurs endroits, JBoss Enterprise BRMS Platform et ses documentations font référence à Guvnor. Drools Guvnor est le nom du projet open-source sur lequel la plate-forme JBoss Enterprise BRMS a été fondée. Il reste des références à Guvnor dans l'API, les URL, et dans les outils de JBoss Developer Studio.

CHAPITRE 2. INSTALLATION

Lire ce chapitre pour apprendre comment installer la plate-forme JBoss Enterprise BRMS.

La plateforme JBoss Enterprise BRMS est supportée par plusieurs configurations de systèmes. Vous trouverez une liste complète de serveurs d'applications pris en charge par Java, de Machines Virtuelles Java (JVM), et de serveurs de bases de données dans

<http://www.jboss.com/products/platforms/brms/supportedconfigurations/>.

En plus du serveur d'applications pris en charge par Java, de JVM, et du système d'exploitation, le déploiement de BRMS dans un environnement de production exige un serveur de bases de données pris en charge, pour le stockage des éléments de règles.

2.1. PACKAGES INFORMATIQUES

JBoss Enterprise BRMS Platform peut être téléchargé sur portail Red Hat Customer Support Portal à partir de <https://access.redhat.com> en naviguant dans **Downloads**, **BRMS Platform**, et en sélectionnant la version. La version actuelle est **5.2.0**. Toutes les instructions ci-dessous utilisent les noms de package de 5.2.0, mais appliquent également des versions antérieures parfois.

La plateforme JBoss Enterprise BRMS est disponible dans deux packages différents.

Package autonome

Ce package autonome contient JBoss Enterprise Web Platform et Jboss Enterprise BRMS Platform déjà déployés, et ne requiert pas de serveur d'applications existants.

Le package autonome est disponible dans la version 5.1.0 et version supérieure.

Si vous téléchargez la version 5.1.0, le package autonome contient :

- JBoss Enterprise Web Platform 5.1.
- Application Web JBoss Enterprise BRMS Platform 5.1.0 déployée dans les profils **par défaut** et de **production**

Si vous téléchargez la version 5.2.0, le package autonome contient :

- JBoss Enterprise Web Platform 5.1.1.
- Application Web JBoss Enterprise BRMS Platform 5.2.0 déployée dans les profils **par défaut** et de **production**.

Consulter [Section 2.2.1, « Installation du package autonome »](#) pour obtenir des instructions sur la façon d'installer ces packages.

Package déployable

Le Package déployable est un fichier ZIP contenant l'application web de la plateforme JBoss Enterprise BRMS en tant que fichier WAR, qui doit être déployé sur un serveur d'applications supportée existant. Le package ZIP contient également des composants facultatifs.

Le package déployable contient les fichiers suivants :

- **jboss-brms-manager.zip** - l'application Web de la plate-forme JBoss Enterprise BRMS.

- **jboss-brms-engine.zip** - bibliothèques du client de la plate-forme JBoss Enterprise BRMS.
- **modeshape.zip** - fichiers nécessaires pour installer le référentiel de ModeShape.



NOTE

modeshape.zip est inclut dans le téléchargement 5.2 en tant qu'aperçu technologique uniquement.

Voir [Section 2.2.2, « Installer le package déployable. »](#) pour obtenir des instructions sur la façon d'installer ce package.

2.2. INSTRUCTIONS D'INSTALLATION

Lire cette section pour apprendre comment installer la plateforme JBoss Enterprise BRMS. Suivre les instructions liées au package qui convient, autonome ou déployable.

Les étapes d'installation optionnelles sont étiquetées comme telles et font référence aux chapitres de ce guide qui contiennent davantage d'informations.

2.2.1. Installation du package autonome

Ce package autonome contient JBoss Enterprise Web Platform et Jboss Enterprise BRMS Platform déjà déployées, comme archive explosée. Suivre la procédure ci-dessous pour l'installer et le configurer.



NOTE

Consulter la documentation JBoss Enterprise Web Platform 5 pour toute configuration supplémentaire requise pour votre environnement particulier,
http://docs.redhat.com/docs/en-US/JBoss_Enterprise_Web_Platform/

Procédure 2.1. Installer le package autonome

1. Download

Télécharger le fichier compressé autonome du portail client Red Hat à partir de <https://access.redhat.com>. Naviguez dans **Downloads**, puis **Plateforme BRMS**, puis, sélectionner la version. La version actuelle est **5.2.0**. Toutes les instructions ci-dessous utilisent les noms de package de 5.2.0, mais appliquent également des versions antérieures parfois.

2. Extrait

Le package autonome contient un seul répertoire contenant l'installation de la plate-forme Web de JBoss Enterprise avec l'application web JBoss Enterprise BRMS Platform déjà déployées. Extrayez le contenu du fichier zip dans le répertoire où il sera installé.

```
[localhost]$ unzip brms-standalone-5.2.0.zip
```

JBoss Enterprise BRMS Platform est déployée à la fois dans les profils de serveur **production** et par défaut : **brms-standalone-5.2.0/jboss-as-web/server/PROFILE/deploy/jboss-brms.war/**

**NOTE**

Ce guide utilise *PROFILE* dans un chemin d'accès de répertoire, comme ci-dessus, pour indiquer les profils **production** ou **par défaut**.

3. Autoriser les utilisateurs

La configuration par défaut utilise le profil d'applications JAAS **brms**. Ce profil stocke des noms d'utilisateurs et des mots de passe dans le fichier **PROFILE/conf/props/brms-users.properties**.

On ajoute des utilisateurs par des entrées au fichier sous le format **nom d'utilisateur=mot de passe**. On peut commenter les entrées en ajoutant le caractère de hachage (#) en début de ligne.

```
#admin=password
jsmith=s@r@hSm1th
tandrews=pp3rrss0nn311
```

Consulter [Section 6.1, « Authentification »](#) pour obtenir des détails sur la façon de changer le profil JAAS qui supporte un système d'authentification comme LDAP.

4. OPTION : Installer le référentiel de ModeShape

ModeShape est un JRC (Java Content Repository / Référentiel de contenu Java) qui peut être utilisé à la place de Apache Jackrabbit. Consulter [Section 2.2.3, « Installer et activer ModeShape »](#) pour obtenir davantage d'informations.

**NOTE**

Modeshape est un aperçu technologique dans BRMS 5.2.

5. OPTION : Activation de Rule Package Signing (signature des règles de package).

La fonctionnalité Rule Signing Package utilise la cryptographie de clés publiques pour veiller à ce que les packages ne soient pas corrompus ou falsifiés au cours du téléchargement. Red Hat recommande que cette fonctionnalité soit activée dans un environnement de production.

Consulter [Section 6.3, « Rule Package Signing »](#) pour étudier ce sujet.

6. Configuration de la base de données

Les configurations par défaut utilisent des bases de données qui ne sont ni adaptées, ni prises en charge pour des environnements de production. Avant de déployer dans un environnement de production, cette configuration doit être changée en un serveur de base de données pris en charge.

Consulter [Chapitre 4, Configuration de la base de données](#).

7. Démarrer le serveur

Pour lancer le serveur à partir de la ligne de commande, utiliser la commande **run.sh** sur un système UNIX ou Linux, ou **run.bat** sur un système Microsoft Windows. Le paramètre **-c** doit être utilisé pour indiquer quel profil de serveur utiliser.

```
[localhost]$ ./run.sh -c production
```

Les scripts **run.sh** et **run.bat** sont situés dans le répertoire **bin**.

Ainsi, le chemin d'accès au répertoire bin du serveur autonome BRMS 5.2.0 est **brms-standalone-5.2.0/jboss-as-web/bin/**.

Consulter la documentation de JBoss Enterprise Web Platform 5 pour obtenir des informations sur les paramètres supplémentaires de lancement du serveur, http://docs.redhat.com/docs/en-US/JBoss_Enterprise_Web_Platform/.

8. Test de connexion

Connectez-vous à l'application web JBoss Enterprise BRMS Platform pour confirmer qu'elle est installée et qu'elle exécute correctement. Voir [Section 2.3, « Première connexion »](#) pour plus de détails.

2.2.2. Installer le package déployable.

Le package déployable contient le logiciel décrit dans [Section 2.1, « Packages informatiques »](#). Suivre la procédure ci-dessous pour installer et configurer le logiciel.

Procédure 2.2. Installer Deployable Package (package déployable)

1. Download

Télécharger le fichier zip de Deployable Package à partir du portail à la Clientèle de Red Hat à l'adresse suivant: <https://access.redhat.com>. Naviguez dans **Downloads**, puis **BRMS Platform**, et sélectionner la version. La version courante est **5.2.0**. Toutes les instructions ci-dessous utilisent les noms de package de 5.2.0, mais s'appliquent également aux versions précédentes si noté.

2. Extraire le logiciel

Extraire les fichiers de déploiement BRMS du package zip.

```
[localhost]$ unzip brms-deployable-5.2.0.zip
```

Cela va créer les fichiers suivants :

- **jboss-brms.engine.zip**
- **jboss-brms.manager.zip**
- **modeshape.zip**



NOTE

modeshape.zip n'est inclus dans le package déployable de BRMS 5.2 qu'en tant qu'aperçu technologique.

Extraire **jboss-brms.war** de **jboss-brms-manager.zip**.

```
[localhost]$ unzip brms-manager.zip
```

Cela va créer un répertoire **jboss-brms.war** qui contient l'application web JBoss Enterprise BRMS Platform sous forme d'archive explosée.

Extraire les bibliothèques client de **jboss-brms-engine.zip** dans un répertoire.

```
[localhost]$ unzip brms-engine.zip -d client
```

Cela crée un répertoire qui contient les fichiers jar de bibliothèques client.

3. Copier dans le serveur d'applications.

Copier l'archive d'applications web explosées dans le répertoire de déploiement de votre serveur d'applications. Veillez à ce que le nom de fichier de l'archive d'applications web explosée soit **jboss-brms.war**. L'exemple ci-dessous démontre que cette opération est effectuée au niveau ligne de commande sur Linux.

```
[localhost]$ cp -rf ~/temp/jboss-brms.war /opt/brms-server/jboss-  
as/server/production/deploy/
```

L'application ne sera pas déployée tant que le serveur d'applications n'est pas lancé.

Copier les fichiers jar du répertoire clients créé dans l'étape précédente dans le serveur d'applications selon les besoins. Veuillez consulter la documentation sur le serveur d'applications pour obtenir davantage d'informations.

4. Autoriser les utilisateurs

La configuration d'authentification par défaut utilise le profil JAAS **jmx-console**. En cas de déploiement dans un serveur d'applications existant, il est possible que ce profil ait déjà été modifié. Dans ce cas, consulter la documentation sur les serveurs d'applications pour obtenir des instructions sur la façon d'ajouter de nouveaux utilisateurs.



NOTE

Tout utilisateur défini dans le serveur d'applications sera en mesure de se connecter à l'installation de JBoss Enterprise BRMS Platform.

Red Hat recommande qu'un système d'authentification séparé soit utilisé pour les utilisateurs de JBoss Enterprise BRMS Platform. Voir [Section 6.1, « Authentification »](#) pour savoir comment modifier le profil JAAS.

La configuration par défaut du profil de la **jmx-console** pour les produits JBoss Platform utilise le module de connexion **UsersRolesLoginModule** pour stocker les noms d'utilisateurs et les mots de passe dans le fichier **PROFILE/conf/props/jmx-console-users.properties**.

On peut ajouter des utilisateurs en ajoutant des entrées à ce fichier sous le format suivant **nom d'utilisateur=mot de passe**. On peut commenter les entrées en ajoutant le caractère de hachage (#) en début de ligne.

```
#admin=sekretz  
jsmith=s@r@hSm1th  
tandrews=p3r50n311
```



NOTE

Ce guide utilise **PROFILE** dans un chemin d'accès de répertoire, comme ci-dessus, pour indiquer les profils **production** ou **par défaut**.

5. OPTION : Activation de Rule Package Signing (signature des règles de package).

La fonctionnalité Rule Signing Package utilise la cryptographie de clés publiques pour veiller à ce que les packages ne soient pas corrompus ou falsifiés au cours du téléchargement. Red Hat recommande que cette fonctionnalité soit activée dans un environnement de production.

Consulter [Section 6.3, « Rule Package Signing »](#) pour étudier ce sujet.

6. Configuration de la base de données

Les configurations par défaut utilisent des bases de données qui ne sont ni adaptées, ni prises en charge pour des environnements de production. Avant de déployer dans un environnement de production, cette configuration doit être changée pour un serveur de base de données pris en charge.

Consulter [Chapitre 4, Configuration de la base de données](#).

7. Démarrer le serveur

Pour lancer le serveur à partir de la ligne de commande, utiliser la commande **run.sh** sur un système UNIX, ou bien exécuter **run.bat** sur un système Microsoft Windows. Le paramètre **-c** doit être utilisé pour spécifier quel profil de serveur utiliser.

```
[localhost]$ ./run.sh -c production
```

Les scripts **run.sh** et **run.bat** sont situés dans le répertoire **bin**.

Ainsi, le chemin d'accès au répertoire bin du serveur autonome BRMS 5.2.0 est **brms-standalone-5.2.0/jboss-as-web/bin/**.

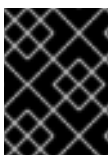
Consulter la documentation de JBoss Enterprise Web Platform 5 pour obtenir des informations sur les paramètres supplémentaires de lancement du serveur, http://docs.redhat.com/docs/en-US/JBoss_Enterprise_Web_Platform/.

8. Connexion

Connectez-vous à l'application web JBoss Enterprise BRMS Platform pour confirmer qu'elle est installée et qu'elle exécute correctement. Voir [Section 2.3, « Première connexion »](#) pour plus de détails.

Consulter [Annexe A, Déployer JBoss BRMS sur EWS](#)

2.2.3. Installer et activer ModeShape



IMPORTANT

Modeshape est fourni dans la version 5.2 de JBoss Enterprise BRMS Platform comme aperçu technologique.

Lire cette section pour apprendre comment remplacer le référentiel par défaut, Apache Jackrabbit, par ModeShape pour l'instance de BRMS autonome.

ModeShape est un JCR (Java Content Repository) qui peut être utilisé à la place d'Apache Jackrabbit. [Procédure 2.3, « Configuration de ModeShape »](#) fournit un aperçu du processus d'installation des services ModeShape et sur la façon de configurer le serveur de JBoss Enterprise BRMS Platform pour utiliser ModeShape.

Les exigences suivantes doivent être remplies avant d'installer ModeShape :

- Le package BRMS autonome a été téléchargé et extrait, voir [Section 2.2.1, « Installation du package autonome »](#) pour obtenir des informations supplémentaires.

**NOTE**

Modeshape est fourni comme aperçu technologique. De ce fait, faire migrer une installation existante vers ModeShape n'est pas pris en charge.

- Apache Ant est installé. Voir [Annexe B, Installer Apache Ant](#) pour les instructions.
- Un serveur de base de données pris en charge est exigé pour le déploiement en production.
 - L'instance de la base de données qui va être utilisée a déjà été créée.
 - Il existe déjà un utilisateur de base de données avec permissions de modifier la base de données.
 - Le fichier JAR du pilote JDBC de la base de données se situe dans le répertoire **lib/** de configuration du serveur.

Consulter <http://www.jboss.com/products/platforms/brms/supportedconfigurations/> pour obtenir une liste complète de serveurs de bases de données pris en charges.

Procédure 2.3. Configuration de ModeShape**1. Téléchargement et Extraction****NOTE**

Cette étape est utile uniquement pour 5.1.0. Le package autonome 5.2.0 est fourni avec ModeShape.

Télécharger le package ModeShape package (modeshape.zip) du portail Red Hat Support Portal à partir de <https://access.redhat.com>

Extraire le répertoire **modeshape** du fichier ZIP du package, et le mettre dans le répertoire d'installation BRMS. Pour la version autonome de JBoss Enterprise BRMS Platform, il s'agit de **brms-standalone-5.1.0**.

```
[localhost]$ unzip modeshape.zip
```

Après l'extraction, le répertoire aura la structure suivante :

```
- brms-standalone-5.2.0
  |- jboss-as-web
  |- mod_cluster
  |- modeshape
  |- picketlink
  |- resteasy
  ` - seam
```

2. Exécuter l'installateur

**NOTE**

Vous aurez besoin des étapes suivantes pour 5.1.0 et 5.2.0, et ces instructions feront référence à 5.2.0 à partir de ce point.

Pour installer ModeShape dans le profil de serveur qui convient, utiliser la ligne de commande pour exécuter **ant**, et quand vous y serez invité, indiquer le profil du serveur où ModeShape sera installé.

```
[localhost modeshape]$ ant
Buildfile: /opt/BRMS/5.1.0/brms-standalone-5.1.0/modeshape/build.xml

determine_home:

set-web-home:

set-as-home:

set-soa-home:

init:
    [echo] JBoss Home is ../jboss-as-web

prompt:
    [input] Enter profile to install ModeShape to: [default]
```

Le script installe tous les services de ModeShape dans le profil de serveur spécifié.

3. Utilisateurs et rôles pour l'installation

Ajouter les comptes d'utilisateur **admin** et **mailman**, et leur assigner les rôles indiqués ci-dessous. La configuration par défaut utilise des fichiers de texte pour stocker les noms d'utilisateur, les mots de passe et les rôles assignés. Les instructions données assument qu'à ce point, la configuration d'autorisation n'a pas été changée. Si elle a été changée, alors il faut ajouter les utilisateurs suivant cette configuration. Consulter [Chapitre 6, Sécurité](#) pour obtenir davantage d'informations.

a. Ajouter les utilisateurs admin et mailman

Ouvrir le fichier **PROFILE/conf/props/brms-users.properties** dans l'éditeur de texte et ajouter les utilisateurs **admin** et **mailman** à ce fichier comme nouvelle ligne avec la syntaxe : **username=password**.

```
admin=s3kr3t5
mailman=53cur3m@1l
```

b. Assigner les rôles

Assigner les rôles indiqués ci-dessous aux deux utilisateurs en ouvrant le fichier **PROFILE/conf/props/brms-roles.properties** dans un éditeur de texte, et en ajoutant une nouvelle ligne pour chaque utilisateur avec la syntaxe : **username=role1, role2, role3**.

L'utilisateur **admin** doit posséder les rôles de **JBossAdmin**, **HttpInvoker**, **user**, et d'**admin**.

L'utilisateur de **mailman** doit posséder les rôles de **JBossAdmin**, **readwrite**.

```
admin=JBossAdmin, HttpInvoker, user, admin
mailman=JBossAdmin, readwrite
```

c. **OPTION : Encoder les mots de passe**

En option, les mots de passe peuvent être stockés en forme encodée et non pas en texte ordinaire. Pour cela, exécuter la commande suivante dans le répertoire du serveur d'application (**jboss-as** or **jboss-as-web**) pour créer la forme encodée du mot de passe, et en remplaçant **PASSWORD** par le mot de passe qui convient.

```
[localhost]$ java -cp client/jboss-logging-
spi.jar:lib/jbosssx.jar
org.jboss.resource.security.SecureIdentityLoginModule PASSWORD
Encoded password: 5f78dc15b9a559cbdf8592078de921bc
```

Puis, remplacer les mots de passe stockés dans **PROFILE/conf/props/brms-users.properties** par les versions encodées.

4. **OPTION : Configuration de la base de données**

La configuration par défaut de Modeshape utilise une base de données qui ne convient, ni n'est supportée dans les environnements de production. Avant de déployer dans un environnement de production, cette configuration doit être changée en serveur de bases de données supportées.

Consulter [Chapitre 4, Configuration de la base de données](#) pour obtenir davantage d'instructions.

5. **Retirer les fichiers JAR Non utilisés**

Supprimer les fichiers JAR suivants du répertoire **server/PROFILE/deploy/jboss-brms.war/WEB-INF/lib/**.

- **jackrabbit-api-2.1.0.jar**
- **jackrabbit-core-2.1.0.jar**
- **jackrabbit-jcr-commons-2.1.0.jar**
- **jackrabbit-spi-2.1.0.jar**
- **hibernate-commons-annotations-3.1.0.GA.jar**
- **hibernate-core-3.3.2.GA.jar**
- **hibernate-validator-3.1.0.GA.jar**
- **lucene-core-2.4.1.jar**
- **jcr-2.0.jar**

6. **Activer ModeShape dans le WAR**

Modifier le fichier **jboss-brms.war/WEB-INF/components.xml** pour retirer la configuration d'Apache Jackrabbit et ajouter la configuration de ModeShape.

Pour retirer la configuration d'Apache Jackrabbit, il faut supprimer ou dé-commenter la section suivante :

```
<property name="properties">
  <key>org.drools.repository.configurator</key>
  <value>org.drools.repository.jackrabbit.JackrabbitRepositoryConfigur
    ator</value>
  <!-- the root directory for the repo storage the directory must
    exist. -->
  <!-- <key>repository.root.directory</key>
    <value>/opt/yourpath</value> -->
</property>
```

La configuration de ModeShape est dé-commentée dans la configuration par défaut. Dé-commenter la configuration comme suit :

```
<property name="properties">
  <key>org.drools.repository.configurator</key>
  <value>org.drools.repository.modeshape.ModeShapeRepositoryConfigurat
    or</value>
  <key>org.modeshape.jcr.URL</key><value>jndi:jcr/local?
    repositoryName=brms</value>
  <key>org.drools.repository.secure.passwords</key><value>>false</value>
  <key>org.drools.repository.admin.password</key><value>admin</value>
  <key>org.drools.repository.mailman.password</key><value>mailman</val
    ue>
</property>
```

7. Définir les mots de passe

ModeShape a besoin de deux comptes d'utilisateur, **admin** et **mailman**. Les mots de passe de ces comptes doivent être spécifiés dans la configuration ModeShape de l'étape précédente.

```
<key>org.drools.repository.secure.passwords</key><value>>false</value>
<key>org.drools.repository.admin.password</key><value>password</valu
e>
<key>org.drools.repository.mailman.password</key><value>password</va
lue>
```

8. Démarrage du serveur

Pour lancer le serveur à partir de la ligne de commande, utiliser la commande **run.sh** d'un système UNIX ou Linux, ou bien **run.bat** sur un système Microsoft Windows. Le paramètre **-c** doit être utilisé pour indiquer quel profil de serveur utiliser.

```
[localhost]$ ./run.sh -c production
```

Les scripts **run.sh** et **run.bat** se situent dans le répertoire **bin**.

Par exemple, le chemin complet vers le répertoire bin du serveur autonome BRMS 5.2.0 est **brms-standalone-5.2.0/jboss-as-web/bin/**.

Consulter la documentation JBoss Enterprise Web Platform 5 pour obtenir des informations sur les paramètres supplémentaires nécessaires pour lancer le serveur, http://docs.redhat.com/docs/en-US/JBoss_Enterprise_Web_Platform/.

9. Connexion

Connectez-vous à l'application web JBoss Enterprise BRMS Platform pour confirmer que l'installation et l'exécution soient au point. Consulter [Section 2.3, « Première connexion »](#) pour obtenir davantage d'informations.

2.3. PREMIÈRE CONNEXION

JBoss Enterprise BRMS Platform fournit une interface utilisateur basée web pour la gestion et le développement des règles. Après une installation réussie, on peut utiliser un navigateur web pour connecter à l'interface.

Procédure 2.4. Connectez-vous à JBoss Enterprise BRMS Platform.

1. Ouvrir un URL dans un navigateur

Ouvrir un navigateur web, et saisir l'URL <http://localhost:8080/jboss-brms/> en remplaçant *localhost* par le nom d'hôte du serveur pour accéder au dialogue de connexion.

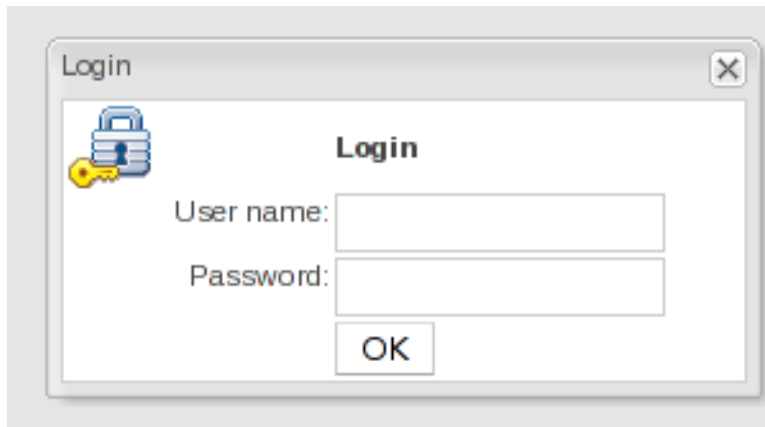


Figure 2.1. Dialogue de connexion

2. Connexion

Saisir les informations d'identification de l'utilisateur

S'il s'agit d'une nouvelle installation, une invite vous demandera si on doit créer un ensemble d'échantillons de règles ou d'autres éléments. Les exemples sont utiles pour la formation, les essais, et à des fins de démonstration.

Une fois que les règles d'échantillonnage auront été installées, la page **Find** sera téléchargée.

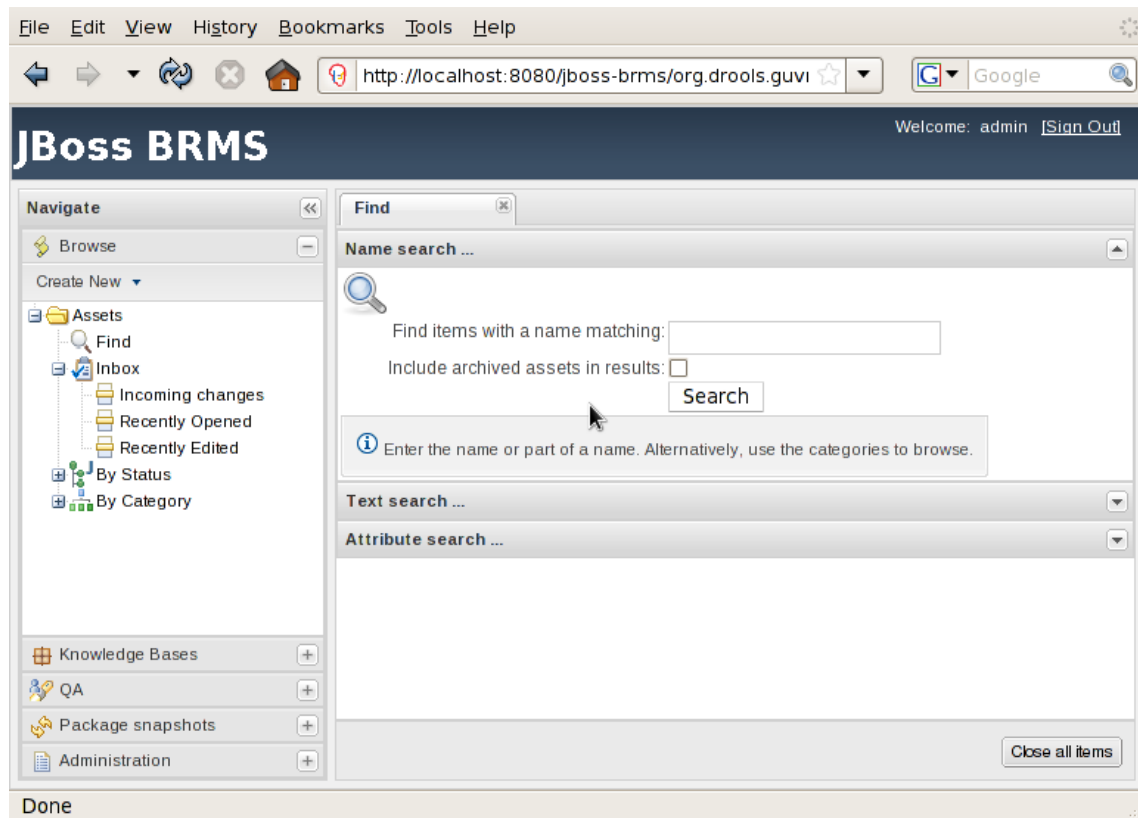


Figure 2.2. Page Find

Si cette procédure ne fonctionne pas comme prévu, consulter [Section 2.4, « Ouvrir une session de dépannage »](#).

2.4. OUVRIR UNE SESSION DE DÉPANNAGE

Voici une liste de problèmes communs que l'on peut rencontrer quand on se connecte à la plateforme JBoss Enterprise BRMS pour la première fois :

J'ai ouvert l' URL dans mon navigateur et il n'y avait pas de page de login :

- Vérifier que le serveur d'applications est en cours d'exécution, et qu'il est bien lancé. Suivant la charge du serveur, les durées de démarrage peuvent varier.
- Vérifier que l'URL est correct pour la configuration du serveur. Le serveur d'applications peut être configuré avec un nom d'hôte ou un port qui varie par rapport à la valeur par défaut.
- Vérifier qu'un pare-feu ne bloque pas l'accès. Si un pare-feu bloque l'accès, on aura besoin de le configurer pour ouvrir l'accès au port sur lequel le serveur exécute.
- Vérifier que toutes les étapes d'installation et de configuration ont été exécutées correctement.

J'ai eu le message "Vous devez configurer un magasin de clés avant de continuer" quand j'ai essayé d'ouvrir une session.

Rule Packaging Signing est activé mais n'est pas configuré correctement.

Consulter [Section 6.3, « Rule Package Signing »](#).

J'ai eu le message "Nom d'utilisateur et Mot de passe erronés" quand j'ouvre une session.

Soit les informations d'authentification sont erronées, soit le système d'authentification n'est pas configuré correctement.

- Vérifier le nom d'utilisateur et le mot de passe. Ces informations d'identification sont sensibles à la casse.
- Vérifier que tout changement apporté à la configuration de l'authentification soit bien correct.

Consulter [Section 6.1, « Authentification »](#) pour obtenir des informations sur la configuration de l'authentification.

Si le problème persiste, vérifier les journaux du serveur. Par défaut, le profil de serveur de **production** a une journalisation limitée pour des raisons de performance. La journalisation a sans doute besoin d'être configurée pour donner davantage d'informations pour le débogage. Red Hat recommande de restaurer la configuration de journalisation d'origine quand le problème est résolu. Voir [Chapitre 7, Ouverture de session](#) pour obtenir des informations sur la configuration de la journalisation.

Si le problème persiste, veuillez contacter Red Hat Support.

CHAPITRE 3. LOCALISATION

L'interface d'utilisateur web de la plateforme d'applications de JBoss Enterprise peut être lue dans plusieurs langues :

- Anglais États-Unis (**en-US**)
- Japonais (**ja-JP**)
- Chinois simplifié (**zh-CN**)

Si aucune langue n'est indiquée, l'anglais US est utilisé par défaut.

Procédure 3.1. Changer la langue de l'interface Web

1. Trouver `index.jsp`

Le fichier `index.jsp` se situe dans le répertoire `jboss-brms.war`.

2. Éditer `index.jsp`

Éditer le fichier `index.jsp` pour spécifier une locale différente. Le fichier `index.jsp` utilise l'anglais US comme langue.

Exemple 3.1. Fichier `index.jsp` par défaut

```
<%
String redirectURL = "org.drools.guvnor.Guvnor/Guvnor.html";
response.sendRedirect(redirectURL);
%>
```

Pour spécifier le japonais comme langue par défaut, éditer le fichier suivant :

Exemple 3.2. `index.jsp` avec japonais comme langue par défaut.

```
<%
String redirectURL = "org.drools.guvnor.Guvnor/Guvnor.html?
locale=ja_JP";
response.sendRedirect(redirectURL);
%>
```

Pour indiquer le chinois simplifié comme langue par défaut, modifier le fichier comme suit :

Exemple 3.3. `index.jsp` avec le chinois simplifié comme langue par défaut.

```
<%
String redirectURL = "org.drools.guvnor.Guvnor/Guvnor.html?
locale=zh_CN";
response.sendRedirect(redirectURL);
%>
```

Les langues prises en charge sont listées dans [Tableau 3.1, « Les URL des langues prises en charge »](#) :

Tableau 3.1. Les URL des langues prises en charge

Langue	Chemin URL
Japonais	<code>org.drools.guvnor.Guvnor/Guvnor.html?locale=ja_JP</code>
Chinois simplifié	<code>org.drools.guvnor.Guvnor/Guvnor.html?locale=zh_CN</code>
Anglais US	<code>org.drools.guvnor.Guvnor/Guvnor.html?locale=en_US</code>

Une langue alternative peut être spécifiée sur la base d'une session en saisissant manuellement l'URL dans le paramètre de locale inclus.

Les URL complets pour les langues prises en charge

Japonais

http://<localhost>:8080/jboss-brms/org.drools.guvnor.Guvnor/Guvnor.html?locale=ja_JP

Chinois simplifié

http://<localhost>:8080/jboss-brms/org.drools.guvnor.Guvnor/Guvnor.html?locale=zh_CN

Anglais US

http://<localhost>:8080/jboss-brms/org.drools.guvnor.Guvnor/Guvnor.html?locale=en_US

CHAPITRE 4. CONFIGURATION DE LA BASE DE DONNÉES

Lire ce chapitre pour apprendre comment configurer les bases de données utilisées par le référentiel de JBoss Enterprise BRMS Platform, changer l'adresse du référentiel, et configurer un RDBMS externe.

La plate-forme JBoss Enterprise BRMS peut soit utiliser Apache Jackrabbit (par défaut), ou Modeshape comme référentiel de contenu. Ces deux référentiels utilisent une base de données pour le stockage, mais chacun d'entre eux est configuré différemment. Les sections suivantes contiennent des instructions sur la façon de configurer chaque base de données du référentiel.



AVERTISSEMENT

Les configurations par défaut d'Apache Jackrabbit et de Modeshape utilisent des bases de données incluses, non prises en charge pour la production, Derby et Hypersonic respectivement. Cette configuration est incluse uniquement à des fins d'évaluation et démonstration et n'est pas prise en charge dans les environnements de production. La configuration d'une base de données supportée est requise pour un déploiement de prise en charge de la production. Consulter <http://www.jboss.com/products/platforms/brms/supportedconfigurations/> pour obtenir une liste des systèmes de bases de données prises en charge.

4.1. CONFIGURATION DE BASE DE DONNÉES DANS APACHE JACKRABBIT

Le référentiel Apache Jackrabbit stocke deux ensembles d'information, *Workspace* et *Versions*. *Workspace* correspond au système de stockage à desseins multiples, qui contient les règles et autres données. *Versions* stocke l'historique de tous les changements à *Workspace*. *Workspace* et *Versions* peuvent être stockés dans des bases de données séparées mais Red Hat recommande d'utiliser une simple base de données pour faciliter la gestion.

Le référentiel de JBoss Enterprise BRMS Platform gère l'accès à ces bases de données par les *persistence managers*. Un gestionnaire de persistance générique est fourni pour les bases de données compatibles-JDBC, ainsi que plusieurs autres pour les implémentations de bases de données spécifiques. Voir <http://wiki.apache.org/jackrabbit/PersistenceManagerFAQ> pour en savoir davantage sur les gestionnaires de persistance, et voir [Annexe C, Exemple de configuration du gestionnaire de persistance](#) pour les exemples.



AVERTISSEMENT

Utiliser la fonctionnalité d'import export (voir [Section 5.2, « Importer et exporter »](#)) pour conserver tout élément stocké dans la base de données existante, avant de modifier la configuration de la base de données.

Veillez à ce que ce qui suit soit disponible ou prêt avant de continuer :

- Base de données vide à utiliser par la plateforme JBoss Enterprise BRMS.
- Compte d'utilisateur de serveur de base de données à utiliser par la plate-forme JBoss Enterprise BRMS. La plateforme JBoss Enterprise BRMS va générer des tableaux de bases de données dans une base de données particulières, s'ils ne sont pas déjà présents. Une fois réalisé, Red Hat recommande de retirer la permission de l'utilisateur de la base de données pour créer les tableaux.
- Le **JAR** de pilote JDBC approprié pour le serveur de base de données.

4.1.1. Configurer la base de données de Workspace

On trouve la configuration de la base de données de Workspace dans le fichier **Workspaces/Default/workspace.xml**. Le fichier **repository.xml** contient les paramètres de configuration par défaut utilisés pour créer les nouveaux packages. Au lieu de mettre à jour le Workspace par défaut déjà existant, il est préférable de le supprimer et de mettre à jour la configuration modèle qui se situe dans **repository.xml**.

Procédure 4.1. Changer la base de données du référentiel.

1. Fermeture

Fermer le serveur de l'application avant d'effectuer ces changements.

2. Installer le JAR du pilote JDBC

Copier le fichier **JAR** qui contient le pilote JDBC dans le répertoire **lib** du serveur, qui pourrait correspondre, par exemple à **jboss-as-web/PROFILE/production/lib/**.

3. Ouvrir repository.xml dans un éditeur de textes.

Ouvrir le fichier **repository.xml** dans un éditeur de texte. Quand on utilise JBoss, ce fichier réside dans le répertoire **bin** du serveur d'applications, à moins qu'une autre adresse n'ait été spécifiée (comme expliqué dans [Section 4.3.1, « Changer l'adresse du référentiel de JackRabbit »](#)).

4. Mise à jour de la configuration de PersistenceManager

Trouver l'élément **<PersistenceManager>**, qui se trouve dans **<Workspace>**.

Par exemple :

```
<PersistenceManager class=
"org.apache.jackrabbit.core.persistence.bundle.DerbyPersistenceManag
er">
  <param name="url"
value="jdbc:derby:${wsp.home}/db;create=true"/>
  <param name="schemaObjectPrefix" value="${wsp.name}_"/>
</PersistenceManager>
```

Remplacer cela par la configuration de la base de données à utiliser pour la base de données. Voir [Annexe C, Exemple de configuration du gestionnaire de persistance](#) pour les exemples.

5. Définir le paramètre schemaObjectPrefix

Cette valeur est indiquée par le préfixe du nom de chaque tableau de base de données géré par le gestionnaire de persistance. (Cela doit être unique dans chaque base de données). La configuration par défaut utilise la variable **\${wsp.name}** qui contient le nom de l'espace de travail en cours.


```
<param name="schemaObjectPrefix" value="${wsp.name}_" />
```

6. Effacer l'ancien espace de travail.

Effacer le répertoire **workspaces/Default/**.

7. Redémarrage

Redémarrer le serveur d'applications.

4.1.2. Configurer la base de données de Versions

JBoss Enterprise BRMS conserve un historique de tous les changements effectués sur la base de données. Cela est conservé dans un lieu de stockage séparé pour empêcher la dégradation au fur et à mesure que le nombre de versions augmente.

Cette zone de stockage a sa propre configuration de gestion des persistances, qui se trouve dans le fichier **repository.xml**. À la différence de la configuration de Workspace, la configuration de stockage de version ici ne consiste pas en un simple modèle; il s'agit de la configuration réelle qui est utilisée.

Procédure 4.2. Modifier la base de données de Versions

1. Fermeture

Fermer le serveur de l'application avant d'effectuer ces changements.

2. Installer le JAR du pilote JDBC

Copier le fichier **JAR** qui contient le pilote JDBC dans le répertoire **lib** du serveur, qui pourrait correspondre, par exemple à **jboss-as/server/PROFILE/lib/**.

3. Ouvrir repository.xml dans un éditeur de textes.

Ouvrir le fichier **repository.xml** dans un éditeur de texte. Quand on utilise JBoss, ce fichier réside dans le répertoire **bin** du serveur d'applications, à moins qu'une autre adresse n'ait été spécifiée (comme expliqué dans [Section 4.3.1, « Changer l'adresse du référentiel de JackRabbit »](#)).

4. Mise à jour de la configuration de PersistenceManager

Trouver l'élément **<PersistenceManager>**, qui se trouve dans **<Workspace>**.

Par exemple :

```
<PersistenceManager class=
"org.apache.jackrabbit.core.persistence.bundle.DerbyPersistenceManag
er">
  <param name="url"
value="jdbc:derby:${rep.home}/version/db;create=true"/>
  <param name="schemaObjectPrefix" value="version_" />
</PersistenceManager>
```

Remplacer cela par la configuration de la base de données à utiliser pour la base de données. Voir [Annexe C, Exemple de configuration du gestionnaire de persistance](#) pour les exemples.

5. Définir le paramètre schemaObjectPrefix

Cette valeur est indiquée par le préfixe qui se trouve sur le nom de chaque tableau de base de données géré par le Gestionnaire de persistance. (Cela doit être unique pour chaque base de données.) Si la même base de données est utilisée pour les deux Versions et pour le datastore,

veillez à ce que le même `schemaObjectPrefix` ne soit pas utilisé pour les deux. (La configuration par défaut utilise **version_**.)

```
<param name="schemaObjectPrefix" value="version_" />
```

6. Redémarrage

Redémarrer le serveur d'applications.

4.1.3. Recherche et Indexation

Fonctionnalité de recherche et d'indexation de JBoss Enterprise BRMS Platform fournie dans *Apache Lucene* (<http://lucene.apache.org/>).

Par défaut, l'index de recherche est conservé dans un système de fichiers local. C'est parce que cela fournit une performance plus rapide. Red Hat ne conseille pas ce paramètre par défaut, à moins que vous ayez un besoin particulier de l'utiliser.

Pour configurer l'adresse de l'index de recherche, modifier l'élément **repository.xml** file's **<SearchIndex>**

```
<SearchIndex class="org.apache.jackrabbit.core.query.lucene.SearchIndex">
  <param name="path" value="${wsp.home}/index"/>
  <param name="extractorPoolSize" value="2"/>
  <param name="supportHighlighting" value="true"/>
</SearchIndex>
```

4.2. CONFIGURATION DE LA DATABASE DE MODESHAPE

La configuration par défaut du référentiel Modeshape utilise un — **ModeShapeBRMSRepo** — JNDI pour le stockage de sources de données, qui est configuré dans le fichier **PROFILE/deploy/modeshape-brms-store-ds.xml**. Modifier ce fichier pour changer la configuration de la source de données, pour pouvoir utiliser la source de données de votre choix, au lieu d' Hypersonic. Ce fichier est un fichier de configuration de source de données de JBoss standard.

Consulter le chapitre de Configurations des sources de données dans la documentation sur le serveur d'applications pour obtenir des détails sur la façon de configurer les sources de données de JBoss.

4.3. CONFIGURATION DU RÉFÉRENTIEL

Lire cette section pour savoir comment configurer un référentiel ou un RDBMS externe.

La JBoss Enterprise BRMS Platform utilise la spécification JRC (Content Repository API for Java) pour stocker et pour suivre les actifs comme les règles. Apache est Jackrabbit dans l'implémentation qui est incluse.

4.3.1. Changer l'adresse du référentiel de JackRabbit

Lire cette section pour savoir comment changer l'adresse dans laquelle Jackrabbit va stocker son référentiel.

Quand l'application JBoss Enterprise BRMS Platform démarre pour la première fois, elle crée un référentiel. À moins de changer la configuration par défaut, ce référentiel va être généré dans le répertoire **bin** du serveur d'applications.

Indiquer une adresse différente en modifiant le fichier de configuration **components.xml** de JBoss Seam pour ce WAR. Sélectionner une adresse sécurisée qui est sauvegardée régulièrement.

Procédure 4.3. Changer la location du référentiel

1. Fermer le serveur d'applications

Stopper le serveur d'applications avant d'effectuer ces changements.

2. Trouver le fichier **components.xml**

Le fichier **components.xml** se situe dans le répertoire **WEB-INF/**, (**deploy/jboss-brms.war/WEB-INF/**.) Il s'agit d'un fichier de configuration standard qui permet la personnalisation de différentes parties de l'application.

3. Trouver l'attribut Key-Value de **repository.root.directory**

Ouvrir le fichier **components.xml** dans un éditeur de texte et trouver l'attribut Key-Value de **repository.root.directory**. Il est contenu dans le composant **repositoryConfiguration** et est "dé-commenté" par défaut.

```
<property name="properties">
  <key>org.drools.repository.configurator</key>

  <value>org.drools.repository.jackrabbit.JackrabbitRepositoryConfigur
  ator</value>
  <!-- the root directory for the repo storage the directory must
  exist. -->
  <!-- <key>repository.root.directory</key>
  <value>/opt/yourpath</value> -->
</property>
```

NOTE

Pour les versions de BRMS antérieures à 5.1, suivre les étapes suivantes :

- Trouver la propriété **homeDirectory**
- Ouvrir le fichier **components.xml** dans un éditeur de textes et trouver la propriété **homeDirectory**. Elle se trouve dans le composant **repositoryConfiguration** et elle est "dé-commentée" par défaut.

```
<component name="repositoryConfiguration">
  <!--
  *** This is for configuring the "home" directory for the
  repository
  storage. the directory must exist. ***
  <property name="homeDirectory">
    /home/michael/RulesRepository_001
  </property>
  -->
</component>
```

4. Mettre à jour l'attribut Key-Value de **repository.root.directory**

Retirer les balises de commentaires qui se trouvent autour de l'attribut de Key-Value `repository.root.directory` et modifier la valeur du chemin d'accès au répertoire où va se trouver le datastore. (Ce répertoire doit déjà exister).

```
<property name="properties">
  <key>org.drools.repository.configurator</key>

  <value>org.drools.repository.jackrabbit.JackrabbitRepositoryConfigur
ator</value>
  <!-- the root directory for the repo storage the directory must
exist. -->
  <key>repository.root.directory</key>
  <value>/opt/brms-standalone-5.2.0/BRMSRulesRepository</value>
</property>
```



NOTE

Pour les versions de BRMS antérieures à 5.1, le code est le suivant :

```
<component name="repositoryConfiguration">
  <property name="homeDirectory">/opt/jboss-soa-
platform/BRMSRulesRepository</property>
</component>
```

5. Déplacer le datastore existant (en option)

La JBoss Enterprise BRMS Platform va créer un nouveau datastore à cette adresse s'il n'y en a pas déjà un. Pour conserver un datastore déjà existant, copier les fichiers existants à une autre adresse avant de redémarrer le serveur d'applications.

6. Redémarrage du serveur d'applications

Redémarrer le serveur d'applications. Si un datastore existant n'a pas été déplacé vers la nouvelle adresse, un nouveau datastore sera créé.

4.3.2. Configurer BRMS pour qu'il utilise un RDBMS externe.

La JBoss Enterprise BRMS Platform peut être configurée pour utiliser un RDBMS externe comme datastore. Le fichier **repository.xml** contient des informations sur l'endroit où les données du référentiel se trouvent. Ce fichier doit être modifié pour installer un RDBMS externe. Ce fichier peut être modifié manuellement ou par l'outil **Repository Configuration** qui se trouve dans l'interface d'utilisateur de BRMS.

Pour éditer manuellement le fichier **repository.xml**, localiser le fichier et ouvrez-le dans un éditeur de texte. Le fichier est annoté par des commentaires qui décrivent les options.

À moins qu'un répertoire d'accueil différent ne soit spécifié (comme décrit dans [Section 4.3.1, « Changer l'adresse du référentiel de JackRabbit »](#)), le fichier **repository.xml** se situera dans le répertoire **bin** du répertoire du serveur d'applications.

Voir [Annexe C, Exemple de configuration du gestionnaire de persistance](#) pour les configurations d'exemples.

Procédure 4.4. Configurer un RDBMS externe par l'outil de configuration du référentiel.

1. Connectez-vous à l'interface utilisateur

Ouvrir un navigateur Web et saisir l'URL <http://localhost:8080/jboss-brms/> en remplaçant *localhost* par le nom d'hôte du serveur pour accéder au dialogue de login.

2. Sélectionner l'outil de configuration du référentiel

À partir du panneau de navigation, sélectionner **Administration**, puis **Configuration Référentiel**.

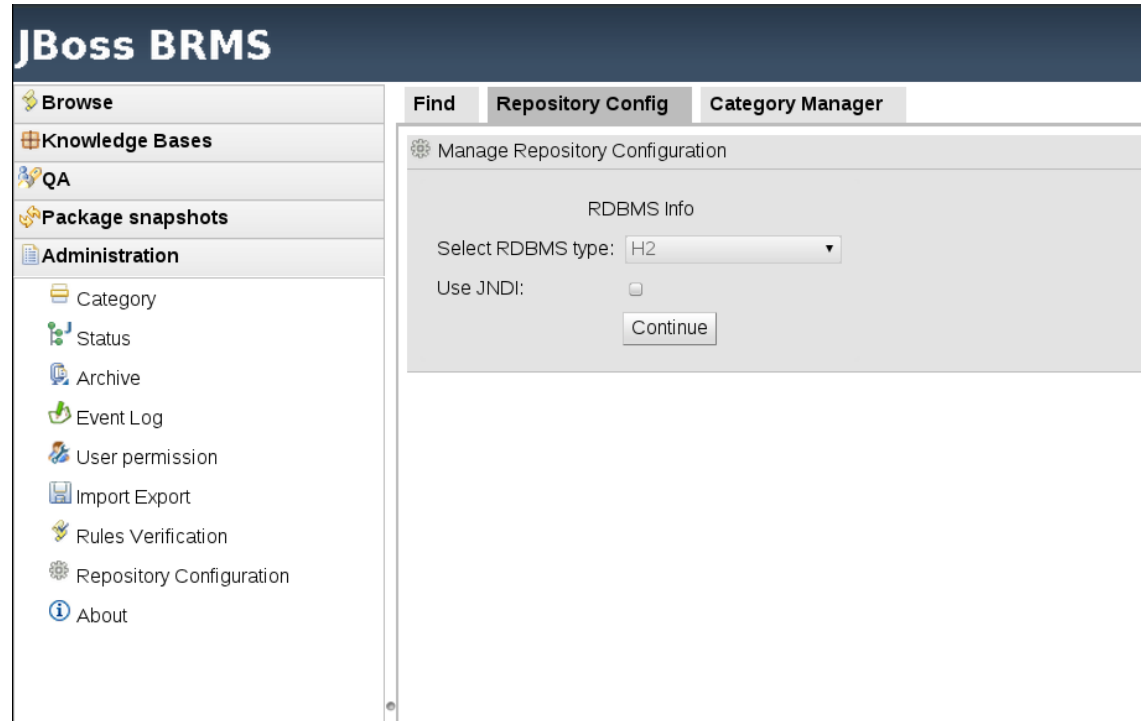


Figure 4.1. Outil de configuration du référentiel

3. Sélectionner le Type RDBMS

À partir du menu déroulant **Select RDBMS type:**, choisir le type RDBMS.

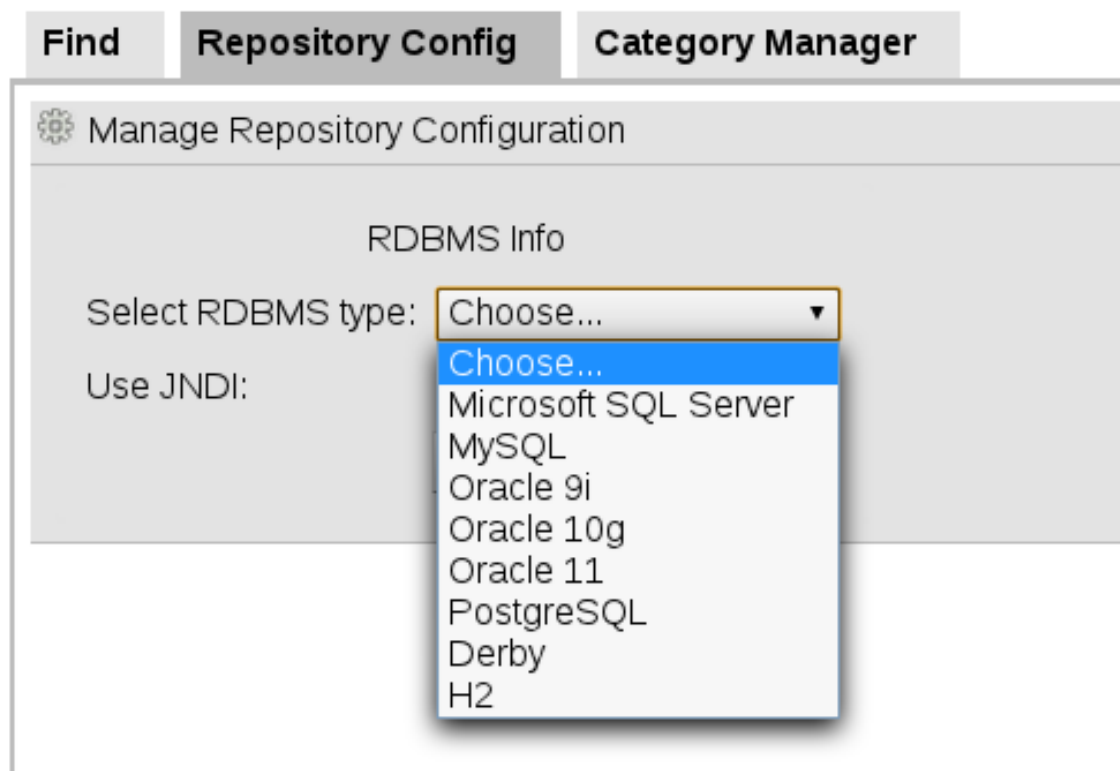


Figure 4.2. Type RDBMS

4. Saisir les informations sur le RDBMS

Sélectionner la case **USE JNDI** si JNDI est configuré.

Sélectionner **Continue** et saisir le nom JNDI configuré dans la source de données déployée.

Si JNDI n'est pas utilisé, saisir les informations RDBMS requises :

- Pilote
- URL
- Utilisateur
- Mot de passe

The screenshot shows a web application interface with three tabs: 'Find', 'Repository Config', and 'Category Manager'. The 'Repository Config' tab is active, displaying a 'Manage Repository Configuration' window. This window has two sections: 'RDBMS Info' and 'MySQL Info'. In the 'RDBMS Info' section, 'Select RDBMS type:' is set to 'MySQL' via a dropdown menu, 'Use JNDI:' is an unchecked checkbox, and there is a 'Continue' button. The 'MySQL Info' section contains four text input fields for 'Driver:', 'URL:', 'User:', and 'Password:', followed by a 'Generate repository config' button.

Figure 4.3. Information RDBMS

5. Générer le fichier de configuration du référentiel

Sélectionner **Generate repository config** pour générer le fichier **repository.xml**.

Vous pouvez, soit couper/coller le texte généré dans un fichier et le sauvegarder dans **repository.xml**, ou bien, vous pouvez sélectionner **Save Configuration** pour télécharger le texte sous forme de fichier. Remplacer le **repository.xml** existant par le nouveau **repository.xml**. À moins qu'un répertoire d'accueil différent ne soit spécifié, (comme décrit dans [Section 4.3.1, « Changer l'adresse du référentiel de JackRabbit »](#)), le fichier **repository.xml** se trouvera dans le répertoire **bin** du serveur d'applications.

CHAPITRE 5. GESTION DES DONNÉES

5.1. SAUVEGARDE DES DONNÉES

La plate-forme BRMS ne fournit pas de solution de sauvegarde. Les sauvegardes de bases de données doivent être faites par des outils fournis par le vendeur de la base de données.

Quand on restaure une sauvegarde, Red Hat recommande tout d'abord d'effacer les index de base de données, pour qu'ils puissent être recréés et optimisés pour les nouvelles données.



AVERTISSEMENT

Ne *pas* utiliser la fonctionnalité d'import ou d'export de référentiel pour effectuer des sauvegardes de référentiel à la place d'utiliser des outils de sauvegarde fournis par le vendeur de base de données. Plusieurs limites, expliquées dans [Section 5.2](#), « [Importer et exporter](#) » le rendent non adapté comme outil de sauvegarde et Red Hat ne le prend pas en charge dans cette capacité.

5.2. IMPORTER ET EXPORTER

La plate-forme JBoss Enterprise BRMS fournit une fonctionnalité import et export standard-JCR à utiliser pour la migration d'une base de données à une autre. Les référentiels exportés sont inscrits dans un fichier XML défini par le standard JCR. On peut accéder à cette fonctionnalité par la section Administration du volet de navigation.



AVERTISSEMENT

La fonctionnalité Import/Export n'est pas sensée être une solution de sauvegarde. Toujours utiliser le système de sauvegarde fourni par le vendeur de bases de données.

Les points suivants doivent être pris en compte pour effectuer une opération d'import/export :

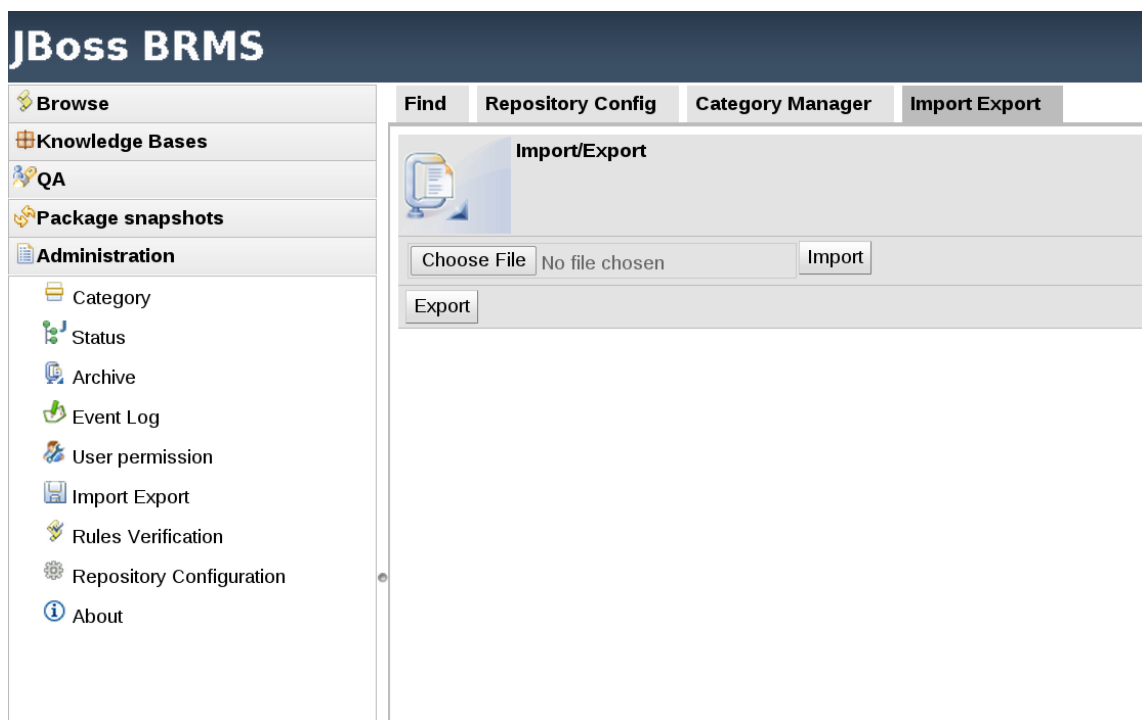
- Quand on importe, tout le contenu de la base de données est retiré.
- Les opérations import et export ne doivent uniquement prendre place quand la base de données n'est pas utilisée.
- La performance dépend à la fois de la taille de la base de données et de la mémoire disponible dans le serveur. L'importation est un processus très coûteux en mémoire.
- L'historique de version du référentiel n'est pas exporté et les règles ont leur attribut Creation Date restauré à la date à laquelle ils ont été importés.

5.3. MIGRER DES DONNÉES ENTRE DES VERSIONS DE BRMS.

Pour faire passer le référentiel d'une ancienne version de **JBoss Business Rules Management System** à la version courante, il faut exporter le référentiel à partir d'une ancienne instance de **JBoss Business Rules Management System** (comme décrit dans [Procédure 5.1, « Exportation du référentiel »](#)), puis d'importer le référentiel dans la version courante, (comme décrit dans [Procédure 5.2, « Importer le référentiel. »](#)).

Procédure 5.1. Exportation du référentiel

1. Sélectionner **Administration** sur la gauche du volet de navigation.
2. Cliquer sur **Import Export**.
3. Cliquer sur **Export**.



Le navigateur téléchargera un fichier compressé qui contient le référentiel dans un fichier XML.

Quelle que soit la taille du référentiel, cette opération prendra quelque temps.



NOTE

Quand on importe un référentiel, il est important de noter que tout contenu du référentiel sera effacé et remplacé par le contenu du fichier XML qui est importé.

Procédure 5.2. Importer le référentiel.

1. Sélectionner **Administration** sur la gauche du volet de navigation.
2. Sélectionner le XML à partir de la machine locale, qui sera téléchargé en cliquant sur **Choose File**, puis cliquer sur **Import**.
3. Cliquer sur **Import**.

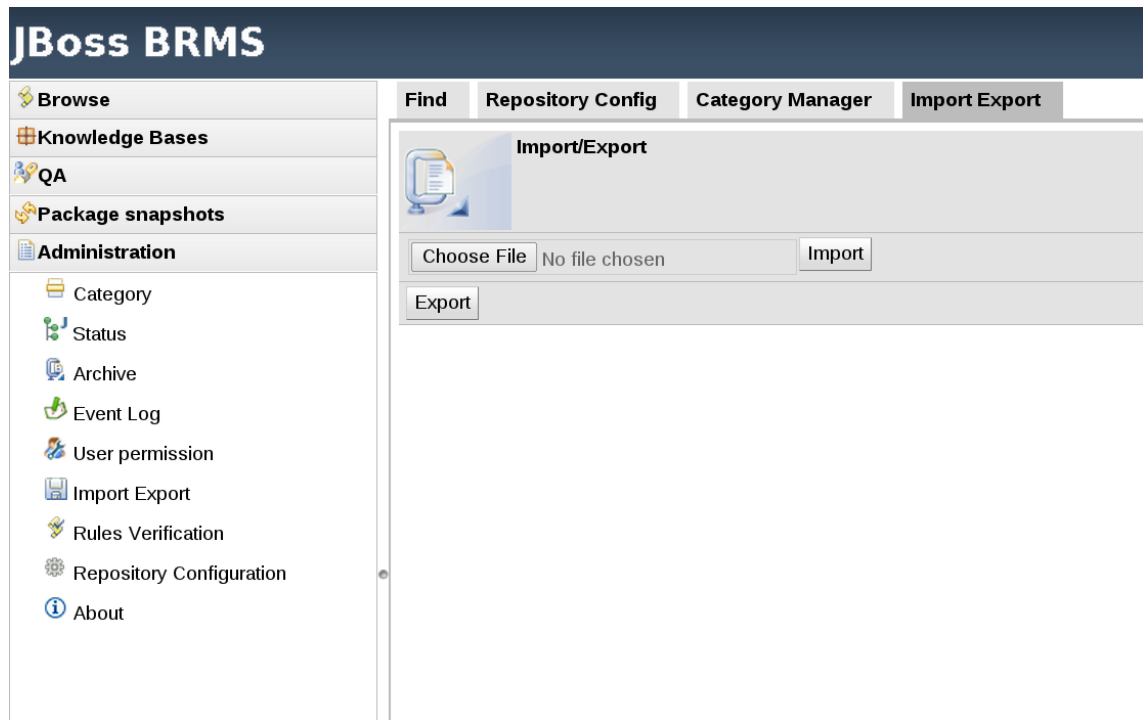


Figure 5.1. Importer et exporter le référentiel

Quand l'importation est terminée, une boîte de dialogue rapportera : «Référentiel de règles importé avec succès. Le navigateur va maintenant réactualiser pour afficher le nouveau contenu».

5.4. REGÉNÉRER LES PACKAGES BINAIRES.

Après la mise à niveau du système **JBoss Business Rules Management System** et après avoir importé un référentiel existant, il est nécessaire de régénérer les packages binaires.

Pour régénérer les packages binaires, sélectionner **Knowledge Bases** (Bases de connaissance) du **Navigation Menu** (Menu de navigation), et cliquer sur **Rebuild all package binaries** (Régénérer tous les binaires du package).

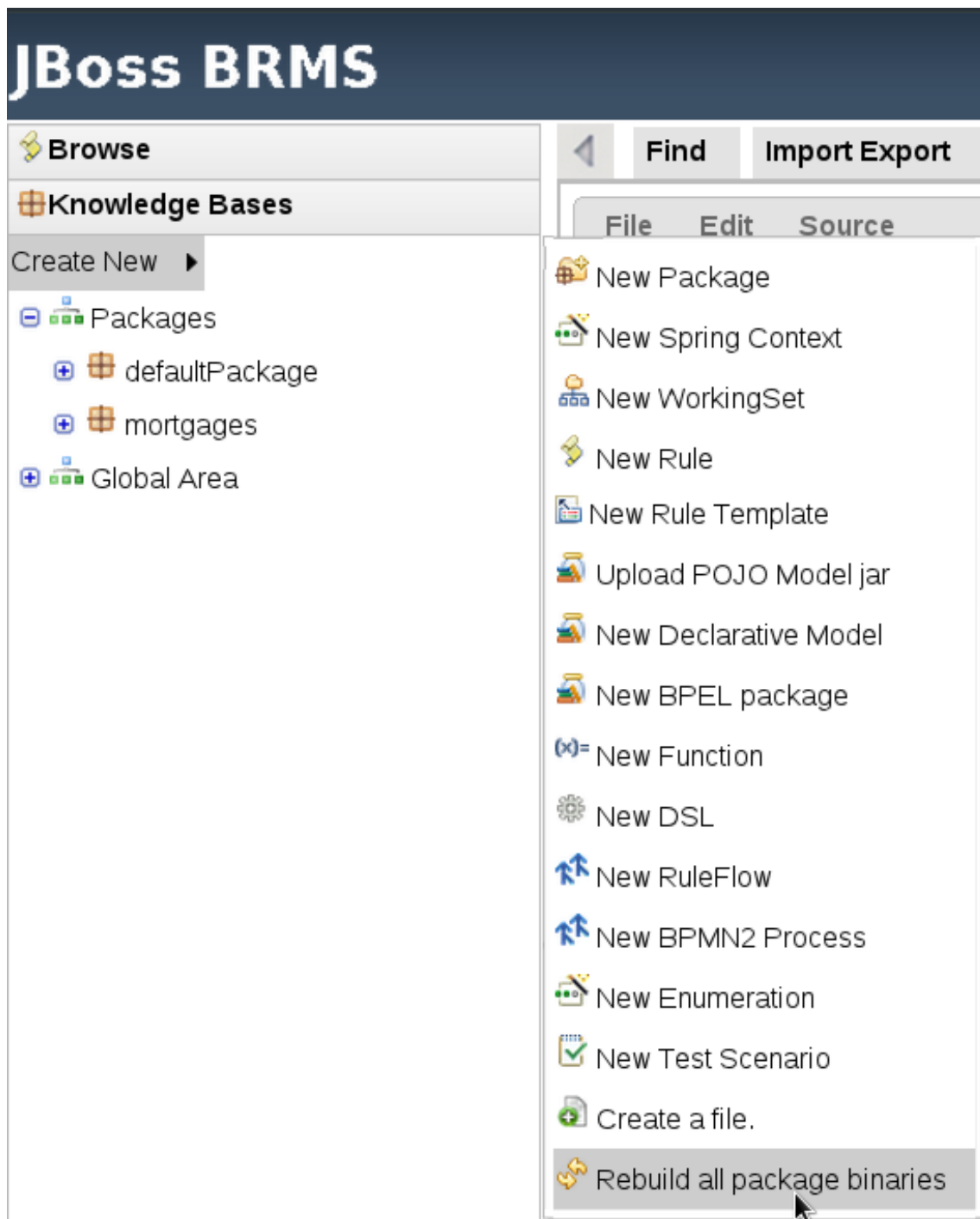


Figure 5.2. Régénération des packages

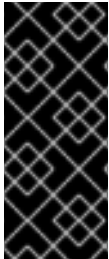
CHAPITRE 6. SÉCURITÉ

Lire ce chapitre pour savoir comment activer et configurer les différents systèmes de sécurité de JBoss Enterprise BRMS Platform. Les différents sujets couverts sont :

- Authentification
- Autorisation
- Rule Package Signing

6.1. AUTHENTIFICATION

La plateforme JBoss Enterprise BRMS Platform utilise *Java Authentication and Authorization Service* pour vérifier les informations d'identification utilisateur. Ce service est fourni par le serveur d'applications et est utilisé pour accéder à un système d'authentification distinct. Le système distinct pourrait être un Lightweight Directory Access Protocol (LDAP), serveur Active Directory, ou base de données JDBC.



IMPORTANT

Lorsque les utilisateurs bénéficient d'accès à l'installation de la plate-forme de JBoss Enterprise BRMS, ils ont effectivement la capacité d'influer la logique métier d'autres applications. Utilisez l'autorisation basée-rôle pour définir ce que chaque utilisateur peut et ne peut pas faire. Lire [Section 6.2, « Autorisation »](#) pour en apprendre davantage sur ce sujet.

Configurer quelle méthode d'authentification utiliser via le fichier **jboss-brms.war/WEB-INF/components.xml**. La configuration par défaut possède plusieurs options «dé-commenter», mais voici un aperçu des paramètres de configuration :

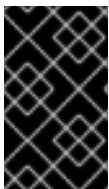
```
<security:identity authenticate-method="#{authenticator.authenticate}"
jaas-config-name="jmx-console"/>
<component name="org.jboss.seam.security.roleBasedPermissionResolver">
  <property name="enableRoleBasedAuthorization">false</property>
</component>
```



NOTE

Le fichier **components.xml** a changé dans BRMS 5.2. Dans BRMS 5.1 et dans les versions précédentes, le fichier ressemble à ceci :

```
<security:identity authenticate-method="#
{authenticator.authenticate}" jaas-config-name="jmx-console"/>
<security:role-based-permission-resolver enable-role-based-
authorization="false"/>
```



IMPORTANT

La configuration utilise les noms de comptes, les mots de passe, et les rôles définis dans la police d'authentification **jmx-console**. Red Hat recommande de modifier cette politique pour l'adapter à votre environnement précis.

Pour configurer l'authentification, suivre les étapes suivantes :

1. Modifier le module de login de JBoss qu'il faut dans le serveur d'applications.
2. Configurer la plateforme JBoss Enterprise BRMS à utiliser pour ce module.



NOTE

De nombreux modules de login de JBoss fournissent un moyen de spécifier un ou plusieurs rôles pour chaque utilisateur. La Plateforme JBoss Enterprise BRMS possède son propre mécanisme de gestion des rôles d'utilisateur.



AVERTISSEMENT

Si *role-based authorization* est activé, tous les utilisateurs ont effectivement le rôle admin. Cela leur donne total accès à JBoss Enterprise BRMS Platform.



IMPORTANT

Red Hat recommande d'activer l'autorisation basée-rôles. Avant de faire cela, Donner le rôle admin à au moins un utilisateur dans l'interface **BRMS Permissions**. Une fois que l'autorisation basée-rôles est activée, seuls les utilisateurs disposant de privilèges d'administrateur pourront exécuter plusieurs tâches administratives, y compris la gestion des rôles d'utilisateur. (Ceci est expliqué plus en détail dans [Section 6.2, « Authorisation »](#).)

6.1.1. Authentication Example: UserRolesLoginModule

Cet exemple illustre l'utilisation du module de login

org.jboss.security.auth.spi.UsersRolesLoginModule qui donne accès à un ensemble de comptes d'utilisateurs dans les fichiers **props/brms-users.properties** et **props/brms-roles.properties**.

Procédure 6.1. Exemple d'authentification : UserRolesLoginModule

1. Veillez à ce que le système d'authentification soit configuré correctement

Ce module de login utilise deux fichiers pour stocker le nom de login, le mot de passe, et les rôles assignés à chaque utilisateur. Créer les fichiers **brms-users.properties** et **brms-roles.properties** dans le répertoire **jboss-as-web/server/PROFILE/conf/props/**, puis spécifier au moins un utilisateur dans **brms-users.properties** en utilisant le format : **username=password**. (le fichier **brms-roles.properties** peut rester vide.)

2. Fermeture

Fermer le serveur d'applications avant de procéder à ces changements.

3. Configurer le module de login de JBoss

Pour configurer les modules de login de JBoss, ouvrir **jboss-as-web/server/PROFILE/conf/login-config.xml** dans un éditeur de texte. Il s'agit d'un fichier XML qui contient un élément **<policy>** avec plusieurs éléments enfant

<application-policy>. Chaque élément d'**<application-policy>** définit un plan d'authentification différent. Ajouter l'extrait XML suivant de **<application-policy>** en tant que nouvel enfant de l'élément **<policy>** :

```
<!--BRMS Platform Security Domain-->
<application-policy name="brms">
  <authentication>
    <login-module
      code="org.jboss.security.auth.spi.UsersRolesLoginModule"
      flag="required">
      <module-option name="usersProperties">
        props/brms-users.properties
      </module-option>
      <module-option name="rolesProperties">
        props/brms-roles.properties
      </module-option>
    </login-module>
  </authentication>
</application-policy>
```

4. Configurer la plateforme BRMS pour utiliser le Module de login

Ouvrir le fichier **jboss-as-web/server/PROFILE/deploy/JBoss-BRMS.war/WEB-INF/components.xml**. Il contient un élément **<components>** ayant plusieurs éléments enfant, y compris **<security:identity>**.

Dé-commentez les éléments **<security:identity>** existants pour empêcher les conflits. Ajouter l'élément **<security:identity>** suivant :

```
<security:identity authenticate-
method="#{authenticator.authenticate}" jaas-config-name="brms"/>
```

La propriété **jaas-config-name** doit être la même que **application-policy**. Si la propriété **application-policy** a été modifiée dans l'étape précédente, modifier la propriété **jaas-config-name** ici pour la faire correspondre.

5. Démarrage à nouveau

Démarrer le serveur d'applications à nouveau.

6.1.2. Exemple d'authentification : LDAP

LDAP est un choix populaire pour les grandes entreprises. Les étapes de configuration de base sont les mêmes que dans l'exemple précédent, mais les détails de configurations seront différents.

Procédure 6.2. Second exemple d'authentification : LDAP

1. Veillez à ce que le serveur LDAP soit configuré correctement

Vérifier que les paramètres de configuration du pare-feu et du réseau n'empêchent pas la communication entre le serveur d'applications et le serveur LDAP.

2. Fermeture

Fermer le serveur d'applications avant de procéder à ces changements.

3. Configurer le module de login de JBoss

Pour configurer les modules de login de JBoss, ouvrir **jboss-as-**

web/server/PROFILE/conf/login-config.xml dans un éditeur de texte. Il s'agit d'un fichier XML qui contient un élément **<policy>** avec plusieurs éléments enfant **<application-policy>**. Chaque élément d'**<application-policy>** définit un plan d'authentification différent. Ajouter l'extrait XML suivant de **<application-policy>** en tant que nouvel enfant de l'élément **<policy>** :

```
<application-policy name="brms">
  <authentication>
    <login-module
      code="org.jboss.security.auth.spi.LdapExtLoginModule"
      flag="required" >
      <module-option name="java.naming.provider.url">
        ldap://ldap.company.com:389
      </module-option>
      <module-option name="bindDN">DEPARTMENT\someadmin</module-
option>
      <module-option name="bindCredential">password</module-option>
      <module-option name="baseCtxDN">cn=Users,dc=company,dc=com
      </module-option>
      <module-option name="baseFilter">(sAMAccountName={0})</module-
option>
      <module-option name="rolesCtxDN">cn=Users,dc=company,dc=com
      </module-option>
      <module-option name="roleFilter">(sAMAccountName={0})</module-
option>
      <module-option name="roleAttributeID">memberOf</module-option>
      <module-option name="roleAttributeIsDN">>true</module-option>
      <module-option name="roleNameAttributeID">cn</module-option>
      <module-option name="roleRecursion">-1</module-option>
      <module-option name="searchScope">ONELEVEL_SCOPE</module-option>
    </login-module>
  </authentication>
</application-policy>
```

Mettre à jour les valeurs de ce fichier de configuration avec celles qui sont appropriées pour votre serveur LDAP.

4. Configurer la platform BRMS pour utiliser le Module de login

Ouvrir le fichier **jboss-as-web/server/PROFILE/deploy/jboss-brms.war/WEB-INF/components.xml**. Il contient un élément **<components>** ayant plusieurs éléments enfants, y compris **<security:identity>**.

Dé-commentez les éléments **<security:identity>** existants pour empêcher les conflits. Ajouter l'élément **<security:identity>** suivant :

```
<security:identity authenticate-method="#
{authenticator.authenticate}" jaas-config-name="brms"/>
```

La propriété **jaas-config-name** doit être la même que **application-policy**. Si la propriété **application-policy** a été modifiée dans l'étape précédente, modifier la propriété **jaas-config-name** ici pour la faire correspondre.

5. Démarrage à nouveau

Démarrer le serveur d'applications à nouveau.



NOTE

Pour en apprendre davantage sur les divers scénarios de configuration LDAP, lire l'information disponible sur ces deux pages web :

<http://www.jboss.org/community/wiki/LdapLoginModule> et
<http://www.jboss.org/community/wiki/LdapExtLoginModule>.

6.2. AUTHORISATION

BRMS utilise une autorisation basée-rôle pour donner des permissions aux utilisateurs. Par défaut, l'autorisation basée-rôle est désactivée, donc elle doit être activée, et des rôles semblables doivent être assignés aux utilisateurs. Si l'autorisation basée-rôle n'est pas attribuée, alors tous les utilisateurs auront des privilèges d'administrateur totaux.

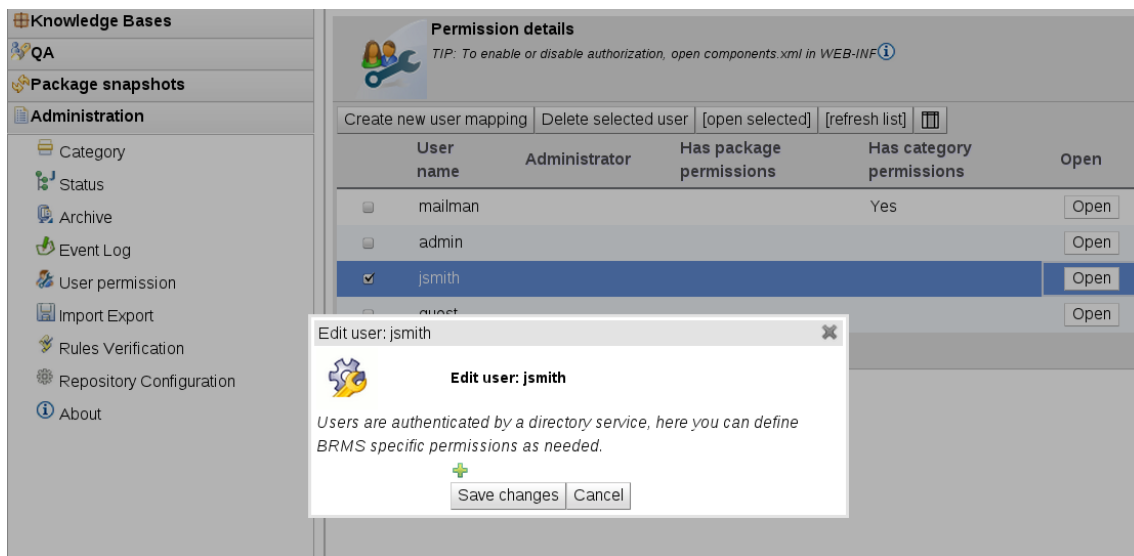


Figure 6.1. Gestion des permissions d'utilisateur

Procédure 6.3. Activer l'autorisation basée-rôle

1. Assigner des privilèges d'administrateur à un utilisateur

Avant d'activer une autorisation basée-rôle, il faut assigner à au moins un utilisateur de confiance le rôle d'administrateur.

Voir [Procédure 6.5, « Gestion des permissions d'utilisateur »](#) pour obtenir davantage d'informations.

2. Fermeture

Fermer le serveur d'applications avant de procéder à ces changements.

3. Ouvrir `components.xml`

Ouvrir le fichier `jboss-as-web/server/PROFILE/deploy/jboss-brms.war/WEB-INF/components.xml` dans un éditeur de texte.

4. Chercher l'élément `<property name="enableRoleBasedAuthorization">`

Dans le fichier par défaut, `components.xml`, cet élément XML est l'enfant de `<components>`.

```
<component
name="org.jboss.seam.security.roleBasedPermissionResolver">
  <property name="enableRoleBasedAuthorization">false</property>
```



```
</component>
```

**NOTE**

Chercher l'XML suivant dans les versions 5.1 et versions précédentes

```
<security:role-based-permission-resolver
    enable-role-based-authorization="false"/>
```

5. Mettre à jour la valeur de l'attribut à «True».

Mettre à jour la valeur de la propriété enable-role-based-authorization à **true** et sauvegarder le fichier.

```
<component
    name="org.jboss.seam.security.roleBasedPermissionResolver">
    <property name="enableRoleBasedAuthorization">true</property>
</component>
```

**NOTE**

Chercher l'XML suivant dans les versions 5.1 et versions précédentes

```
<security:role-based-permission-resolver
    enable-role-based-authorization="true"/>
```

6. Démarrage à nouveau

Démarrer le serveur d'applications à nouveau.

La plateforme JBoss Enterprise ne parvient pas à identifier les utilisateurs. Seuls les utilisateurs BRMS seront visibles dans les Permissions d'utilisateur de plate-forme JBoss Enterprise BRMS.

Procédure 6.4. Ajouter un nouvel utilisateur BRMS**1. Sélectionner les informations sur les permissions**

Sélectionner **Administration** à partir du menu de **navigation**, puis sélectionner **User Permissions**. (Permissions Utilisateur)

2. Ajouter le mappage d'utilisateur

Cliquer sur le bouton **Create new user mapping** (Créer un nouveau mappage d'utilisateur). Saisir le nom d'utilisateur dans la boîte de dialogue, et cliquer sur **OK**.

**NOTE**

Le nom d'utilisateur indiqué pour le rôle doit correspondre à un nom d'utilisateur du service d'authentification ou il ne fonctionnera pas.

3. Ajouter les permissions

Voir [Procédure 6.5, « Gestion des permissions d'utilisateur »](#) pour obtenir davantage d'informations.

Procédure 6.5. Gestion des permissions d'utilisateur

1. Sélectionner les permissions d'utilisateur

Sélectionner **Administration** à partir du **menu de navigation**, puis sélectionner **User Permissions**. (Permissions Utilisateur)

2. Sélectionner l'utilisateur

Sélectionner **Open** qui se trouve à côté du nom d'utilisateur.

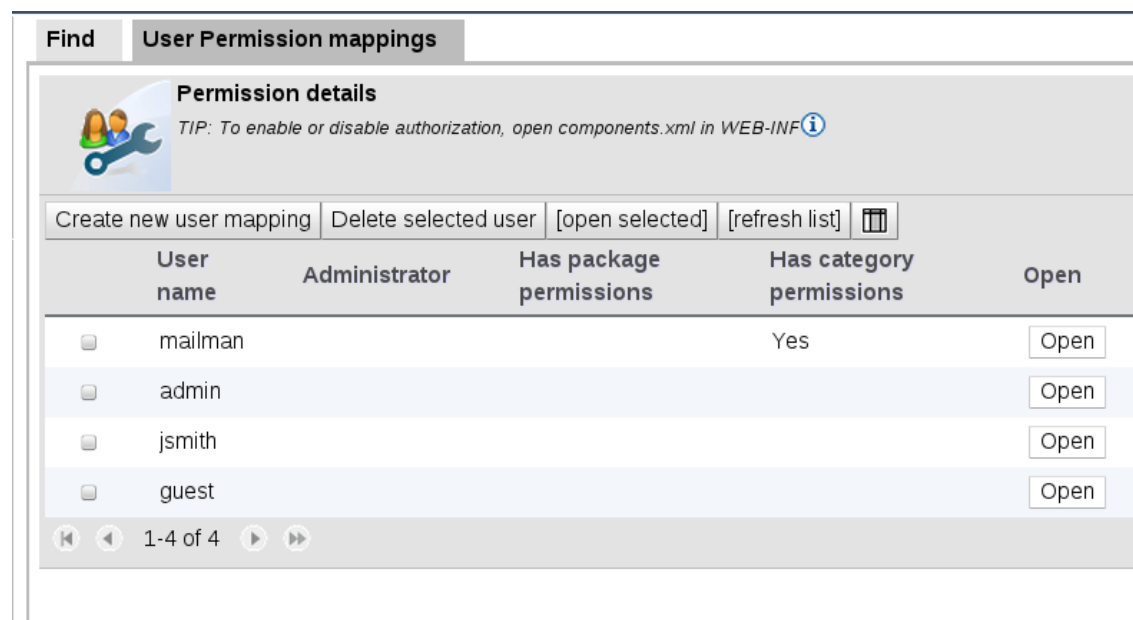


Figure 6.2. Détails Permissions

3. Assigner des permissions à l'utilisateur

Cliquer sur l'icône pour ajouter des permissions, et sélectionner les permissions qui conviennent à partir du menu déroulant de **Permission type**. Appuyez sur **OK** pour confirmer.

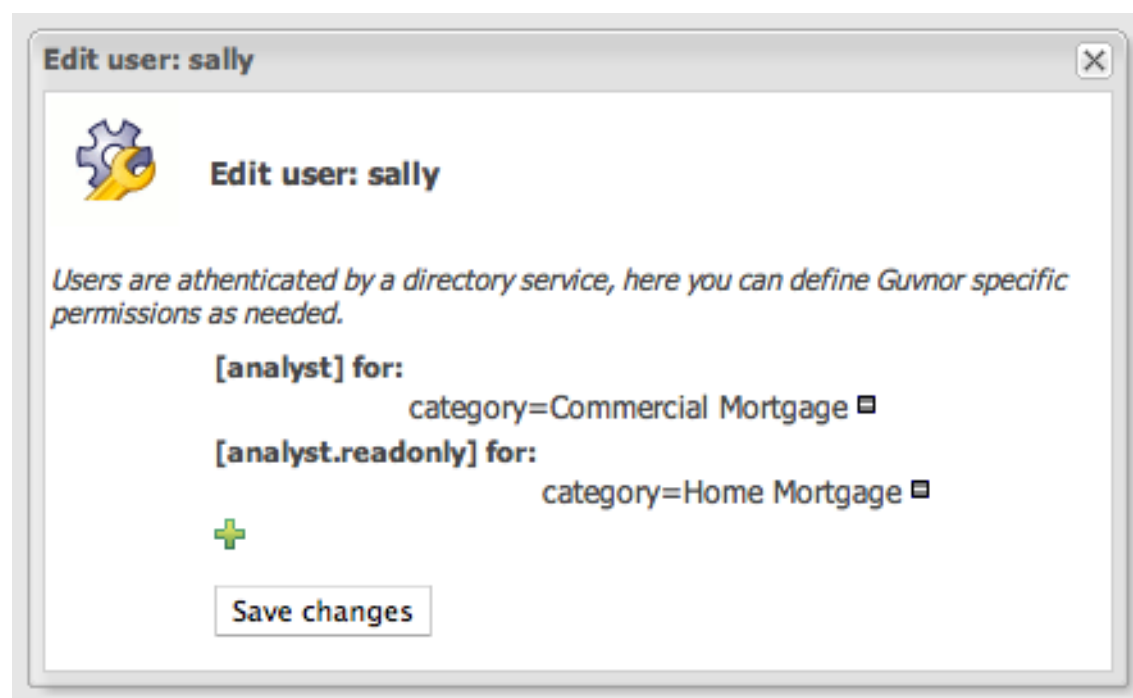


Figure 6.3. Modifier les permissions d'utilisateur

4. Retirer les permissions d'utilisateur

Cliquez sur l'icône moins à côté de la permission supprimée, puis cliquez sur OK pour confirmer.

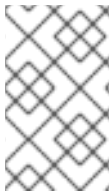
Les utilisateurs qui possèdent le rôle d'administrateur pourront modifier les rôles et les permissions des autres.

Les utilisateurs peuvent recevoir un ou plusieurs rôles suivants :

- admin
- analyst
- package

Les utilisateurs à qui on a confié le rôle d'admin ont accès à toutes les domaines de JBoss Enterprise BRMS Platform.

Les permissions Analyst sont destinées aux utilisateurs qui sont responsables de la maintenance des ressources de règles. Ce niveau de permission est destiné aux développeurs et aux analystes d'entreprise.



NOTE

Quand vous octroyez des permission d'analyste à un utilisateur, on vous demandera de préciser une *category*. Les catégories représentent un façon de grouper les règles indépendamment des packages de base de connaissance auxquels ils appartiennent.



NOTE

Il existe également un permission d'analyste lecture-seule pour les situations telles que lorsqu'un analyste est responsable de maintenir des règles dans une catégorie, mais a besoin d'être en mesure d'en examiner d'autres dans une autre catégorie.



IMPORTANT

Un utilisateur qui ne possède que des catégories de permission ne sera pas en mesure de voir les packages ou leurs détails : ils ne pourront apercevoir que les **Simple Categories**.

Quand vous assignez le rôle, vous êtes invités à sélectionner un package auquel les permissions vont se limiter. Il existe trois différentes sortes de permissions de package parmi lesquelles choisir quand on en assigne une à un utilisateur :

Administrateur de package

La permission d'administrateur de package donne un contrôle total sur un package particulier, y compris le droit de la déployer. La permission d' Administrateur de package ne donne aucun droit administratif à aucune partie de la plateforme JBoss Enterprise BRMS.

Développeur de package

Le développeur de package permet aux utilisateurs de créer et de modifier des items à l'intérieur du package spécifié. Cela inclut la possibilité de créer et de mener des tests, mais n'inclut pas le droit de déployer le package.

Package Lecture-seule

La permission de package lecture-seule ressemble à la permission analyste lecture-seule, mais donne accès à un package, à la place d'une catégorie.

6.3. RULE PACKAGE SIGNING

Lire cette section pour comprendre *Rule Packaging Signing* et la configuration de *keystore*.

La fonctionnalité *Rule Package Signing* veille à ce que les packages ne soient pas corrompus ou falsifiés au cours du téléchargement du serveur de la plateforme BRMS vers les applications du client. Red Hat recommande que cette fonctionnalité soit activée dans un environnement de production.



IMPORTANT

Red Hat recommande fortement que *Rule Package Signing* soit activé dans les environnements de production.

Rule Packaging Signing est implémentée à l'aide de *Public Key Cryptography*. La commande `JDK keytool` est utilisée pour créer une clé privée et un certificat numérique public correspondant. Les packages signés avec une clé privée peuvent uniquement être vérifiés avec le certificat correspondant. La clé privée est stockée dans un fichier appelé *keystore* et est utilisée par le serveur pour signer automatiquement chaque paquet. Le certificat public est mis à la disposition de chaque application client dans un *keystore* appelé un *truststore*. Le certificat du *truststore* est utilisé pour vérifier l'authenticité des paquets signés. Les packages de règles qui sont endommagés ou modifiés au cours du téléchargement seront rejetés par le client parce que la signature ne correspondra n'est plus au certificat.

La procédure ci-dessous décrit le processus de configuration du serveur pour *Rule Package Signing*.

Pour ce process, il faut :

- Créer une clé de signature privée et un certificat digital public correspondant.
- Rendez la clé de signature privée et le certificat digital disponibles dans le serveur dans les *keystores*.
- Configurer le serveur pour qu'il utilise les *keystores*.

Procédure 6.6. Configurer *Rule Package Signing*

1. Créer un *Keystore* privé

Utiliser la commande `keytool` pour créer le *keystore* privé :

```
keytool -genkey -alias ALIAS -keyalg RSA -keystore PRIVATE.keystore
```

Le paramètre **-alias** indique le nom utilisé pour relier les entités reliées dans le *keystore*. Utiliser le même alias pour chacune de ces étapes. L'alias n'est pas sensible à la casse. Le paramètre **-keystore** fournit le nom du fichier qui sera créé pour contenir la clé privée.

`keytool` vous invitera à vous identifier et à fournir deux mots de passe. Le premier mot de passe, le *keystore password*, sécurise le *keystore*. Le second mot de passe, le mot de passe clé, sécurise la clé qui est en cours de création.

```
[localhost]$ keytool -genkey -alias BRMSKey -keyalg RSA -keystore  
PrivateBRMS.keystore  
Enter keystore password:
```

```

Re-enter new password:
What is your first and last name?
[Unknown]: John Smith
What is the name of your organizational unit?
[Unknown]: Accounts
What is the name of your organization?
[Unknown]: ACME INC
What is the name of your City or Locality?
[Unknown]: Captital City
What is the name of your State or Province?
[Unknown]: CC
What is the two-letter country code for this unit?
[Unknown]: US
Is CN=John Smith, OU=Accounts, O=ACME INC, L=Captial City, ST=CC,
C=US correct?
[no]: yes
Enter key password for <BRMSKey>
(RETURN if same as keystore password):
Re-enter new password:

```

2. Créer un Certificat digital

Utiliser la commande **keytool** pour créer un certificat digital :

```

keytool -export -alias ALIAS -file CERTIFICATE.crt -keystore
PRIVATE.keystore

```

Utiliser le même alias et le même keystore que pour l'étape précédente. Le paramètre **-file** est le nom de fichier du nouveau certificat qui va être créé. Le paramètre **-keystore** fournit le nom de fichier du keystore privé.

Saisir le mot de passe du keystore quand vous y êtes invité.

```

[localhost]$ keytool -export -alias BRMSKey -file BRMSKey.crt -
keystore PrivateBRMS.keystore
Enter keystore password:
Certificate stored in file <BRMSKey.crt>

```

3. Importer le Certificat digital dans le Truststore

Utiliser la commande **keytool** pour importer le certificat digital dans un keystore :

```

keytool -import -alias ALIAS -file CERTIFICATE.crt -keystore
PUBLIC.keystore

```

Cela va créer un nouveau keystore, le truststore, qui contient le certificat digital. Le truststore rend le certificat digital disponible aux applications clients.

```

[localhost]$ keytool -import -alias BRMSKey -file BRMSKey.crt -
keystore PublicBRMS.keystore
Enter keystore password:
Re-enter new password:
Owner: CN=John Smith, OU=Accounts, O=ACME INC, L=Captial City,
ST=CC, C=US
Issuer: CN=John Smith, OU=Accounts, O=ACME INC, L=Captial City,
ST=CC, C=US

```

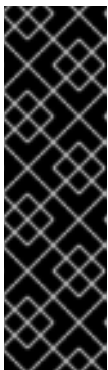
```

Serial number: 4ca0021b
Valid from: Sun Sep 26 22:31:55 EDT 2010 until: Sat Dec 25 21:31:55
EST 2010
Certificate fingerprints:
    MD5: 31:1D:1B:98:59:CC:0E:3C:3F:57:01:C2:FE:F2:6D:C9
    SHA1: 4C:26:52:CA:0A:92:CC:7A:86:04:50:53:80:94:2A:4F:82:6F:53:AD
Signature algorithm name: SHA1withRSA
Version: 3
Trust this certificate? [no]: yes
Certificate was added to keystore

```

4. Déplacer le keystore privé dans une location sécurisée

Le keystore sécurisé a besoin d'être gardé dans une location sécurisée à laquelle le serveur de la plateforme JBoss Enterprise BRMS peut accéder. Il pourrait s'agir de la même machine ou d'une location de réseau sécurisée qui est disponible pour cette machine.



IMPORTANT

La plateforme JBoss Enterprise BRMS Platform n'est pas en mesure de fournir des informations d'identification aux ressources de réseau. Si le keystore privé se trouve dans une location de réseau sécurisée, alors toute procédure d'authentification devra avoir lieu pour le compte du serveur JBoss Enterprise BRMS, pour rendre le keystore privé disponible. Par exemple, le système d'exploitation peut authentifier et monter un fichier partagé qui contient le keystore privé comme répertoire local auquel le serveur de JBoss Enterprise BRMS Platform peut accéder.

5. Déplacer le Truststore vers une location publique

Le truststore a besoin d'être accessible aux applications client. Cela est possible en mettant le truststore sur un réseau en commun ou en l'hébergeant sur un webserveur.

6. Définir les propriétés de sérialisation de Drools.

Les propriétés système de sérialisation de Drools ont besoin d'être configurées sur le serveur. Il s'agit des propriétés qui stockent des informations requises pour accéder aux keystores. Comme JBoss Enterprise BRMS Platform contient également des composants, les propriétés de truststore et du keystore privé devront être définies sur le serveur.

Consulter [Section 6.3.1, « Définir les propriétés de sérialisation »](#) pour obtenir des détails sur l'endroit où définir les propriétés.

Les propriétés qui sont besoin d'être définies sont les suivantes :

- **drools.serialization.sign** - indique si la signature est activée. Doit être défini sur **true**.
- **drools.serialization.private.keyStoreURL** - URL où le keystore privé se situe
- **drools.serialization.private.keyStorePwd** - le mot de passe du keystore
- **drools.serialization.private.keyAlias** - alias utilisé pour créer le keystore
- **drools.serialization.private.keyPwd** - le mot de passe clé
- **drools.serialization.public.keyStoreURL** - URL où se situe le truststore

- `drools.serialization.public.keyStorePwd` - le mot de passe pour le truststore

7. Encrypter les informations d'identification du keystore

Le mot de passe du keystore est actuellement stocké en texte ordinaire.

Consulter <https://access.redhat.com/kb/docs/DOC-47247> pour obtenir des instructions pour masquer les informations d'authentification du keystore.

- Consulter le Guide d'utilisateur de BRMS pour obtenir des instructions sur la façon de configurer les Rule Packages signés.

6.3.1. Définir les propriétés de sérialisation

Les propriétés système des informations d'identification Keystore peuvent être définies à plusieurs endroits, comme décrit ci-dessous. Les propriétés doivent uniquement être définies dans un seul endroit et seront disponibles à toutes les applications s'exécutant sur la même instance de serveur d'application, indépendamment d'où elles sont définies.

JBoss Properties Service

Pour configurer les propriétés de JBoss Properties Service, ajouter la configuration de beans gérés suivante au fichier `/server/PROFILE/deploy/properties-service.xml`, en remplaçant les valeurs d'exemple par celles qui conviennent pour votre système.

```
<mbean code="org.jboss.varia.property.SystemPropertiesService"
      name="jboss:type=Service,name=SystemProperties">
  <attribute name="Properties">
    # Drools Security Serialization specific properties
    drools.serialization.sign=true

    drools.serialization.private.keyStoreURL=file:///opt/secure/PrivateBRMS.
    keystore
    drools.serialization.private.keyStorePwd=storepassgoeshere
    drools.serialization.private.keyAlias=BRMSKey
    drools.serialization.private.keyPwd=keypassgoeshere

    drools.serialization.public.keyStoreURL=file:///opt/public/PublicBRMS.ke
    ystore
    drools.serialization.public.keyStorePwd=keypassgoeshere
  </attribute>
</mbean>
```

jboss-brm.war properties file

Pour configurer les propriétés du fichier de propriétés jboss-brms.war properties, ajouter le code suivant au fichier `jboss-brms.war/WEB-INF/classes/preferences.properties`.

```
drools.serialization.sign=true
drools.serialization.private.keyStoreURL=file:///opt/secure/PrivateBRMS.
keystore
drools.serialization.private.keyStorePwd=storepassgoeshere
drools.serialization.private.keyAlias=BRMSKey
drools.serialization.private.keyPwd=keypassgoeshere
drools.serialization.public.keyStoreURL=file:///opt/public/PublicBRMS.ke
ystore
drools.serialization.public.keyStorePwd=keypassgoeshere
```

CHAPITRE 7. OUVERTURE DE SESSION

L'application web de la plateforme JBoss Enterprise BRMS comprend une fonctionnalité donnée par **log4j**.

La configuration saisie dans **WAR, WEB-INF/classes/log4j.xml** envoie tous les messages dans **STDOUT**. Une fois déployée dans le profil de serveur **production**, les messages de journalisation sont ajoutés au fichier log du serveur **PROFILE/log/server.log**. Une fois déployée dans le profil de serveur par **défaut**, ces messages seront affichés sur la console du serveur.

Pour modifier le comportement de journalisation, modifier la configuration de **log4j** pour le profil de serveur suivant : **jboss-as-web/server/PROFILE/conf/jboss-log4j.xml**.

Ajouter l'XML suivant à la configuration **log4j** du profil de serveur par **défaut** afin de créer une nouvelle catégorie qui redirige tous les messages STDOUT vers le fichier de journalisation du serveur :

```
<category name="STDOUT" additivity="false">
  <priority value="INFO" />
  <appender-ref ref="FILE"/>
</category>
```



NOTE

Voir la documentation de JBoss Enterprise Application Platform pour en apprendre davantage sur la journalisation.

CHAPITRE 8. PERSONNALISATION

8.1. PERSONNALISER L'INTERFACE UTILISATEUR

L'interface d'utilisateur BRMS est produite dynamiquement par le *GWT framework*. L'apparence de l'interface utilisateur peut être personnalisée à des fins de valorisation d'une marque ou dans un but d'intégration, en modifiant les images et les feuilles de style CSS.

Les fichiers **.css** et certaines images se trouvent dans le répertoire **jboss-brms.war/org.drools.guvnor.Guvnor/**. (On peut trouver le reste des images dans le sous-répertoire **images**.) Pour y accéder, le fichier WAR doit être déployé dans une archive explosée, décrite dans [Chapitre 2, Installation](#).

Modifier ou remplacer les images et les fichiers **CSS** et ne pas modifier les noms de fichiers. Si vous rencontrez des problèmes, restaurer les versions originales des fichiers de l'archive **WAR**.



NOTE

Red Hat recommande d'ajouter tout fichier modifié dans un système de contrôle de version pour faciliter la maintenance.

Le changement le plus commun consiste à remplacer les images de marque, c'est à dire le logo qui se trouve en haut de l'écran et l'icône « sites préférés» (**hdrlogo_brms.gif** et **drools.gif** respectivement.) ^[1]

Le fichier **Guvnor.css** contrôle le style général des éléments de la page.

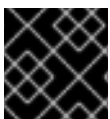


AVERTISSEMENT

Les composants GWT utilisent plusieurs fichiers **CSS** supplémentaires. Ne pas les changer.

Personnaliser les URL utilisés par la plate-forme BRMS, en éditant le descripteur de déploiement, c'est à dire **jboss-brms.war/WEB-INF/web.xml**. Utiliser le même processus que pour toute autre application web Java.

8.2. PERSONNALISER LES SÉLECTEURS POUR LA GÉNÉRATION DES PACKAGES.



IMPORTANT

Il s'agit d'un aperçu technologique de fonctionnalité uniquement.

Sélecteurs en tant que fonctionnalité optionnelle. Quand on construit des packages de base de connaissance en utilisant la fonctionnalité **Packages**, on peut spécifier un *selector*. Un sélecteur filtre la liste des règles figurant dans le package. Saisir le nom du sélecteur, comme configuré sur le serveur,

dans la boîte de dialogue de **selector**.



NOTE

Pour configurer un sélecteur, déployer le **WAR** BRMS en tant qu' «archive explosée». Cela est nécessaire pour pouvoir effectuer une personnalisation ou une tâche de configuration. Davantage d'explications dans [Chapitre 2, Installation](#).

Localiser le fichier **selectors.properties**. Dans ce fichier, il y a des détails sur la façon de configurer un sélecteur personnalisé. Vous pouvez également ajouter un nouveau fichier **selectors.properties** au chemin de classes du système. Les options consistent à utiliser un fichier **DRL** (*Drools Rule Language*), ou bien le nom d'une classe qui implémente l'interface **AssetSelector**. (Il existe déjà un exemple du fichier **DRL** dans le fichier **selectors.properties**.)

Veillez à ce que chaque sélecteur possède un nom unique dans ce fichier de propriétés. Ce sont les noms qui seront utilisés quand on génère des packages de bases de connaissance.

[1] L'icône de «sites préférés» est affichée à différents endroits, suivant le navigateur ou le système d'exploitation. On le voit normalement sur la barre d'adresses du navigateur, le menu marque-pages ou le menu favoris, la barre des titres Windows et comme icône pour la raccourci URL de bureau.

CHAPITRE 9. SURVEILLANCE

JBoss Operations Network est la suite de gestion de système de middleware de JBoss et peut être utilisée pour surveiller **JBoss Business Rules Management System**.

JBoss Operations Network est disponible à partir de <https://access.redhat.com>. Veuillez consulter *JBoss Operations Network Installation Guide* pour les instructions d'installation.

9.1. INSTALLATION DU PLUG-IN JBOSS OPERATIONS NETWORK

Le plug-in BRMS JBoss ON peut être installé par la ligne de commande ou par l'interface d'utilisateur web **JBoss ON**.

Procédure 9.1. Installer le plug-in JBoss Operations Network par la ligne de commandes.

1. Télécharger le plug-in JBoss ON

Télécharger le plug-in **JBoss ON** à partir du [Customer Portail](#) (Portail Clientèle)

- a. Dans le portail de support à la clientèle, cliquer sur **Downloads**, puis **JBoss Operations Network**, puis sélectionner **JBoss ON for BRMS**
- b. Cliquer sur **Download** pour commencer le téléchargement.

2. Décompresser le téléchargement du plug-in.

Décompresser le fichier jar du fichier zip téléchargé :

```
unzip jon-plugin-pack-brms-2.4.1.GA.zip
```

3. Installer le plug-in

Le plug-in a besoin d'être installé dans le répertoire de serveur **JBoss ON**. Déplacer le fichier jar décompressé dans l'étape précédente dans le répertoire du niveau supérieur, c'est à dire, sur Red Hat Enterprise Linux avec l'installation **JBoss ON** dans le répertoire **/opt/**.

```
mv rhq-brms-plugin-5.2.0.jar /opt/jon-server-2.4.0.GA1/rhq-brms-  
plugin-5.2.0.jar
```

4. Redémarrer le serveur JBoss ON.

Procédure 9.2. Installer le plug-in JBoss Operations Network via l'interface Web JBoss ON.

1. Télécharger le plug-in JBoss ON

Télécharger le plug-in **JBoss ON** à partir du [Customer Portail](#) (Portail Clientèle)

- a. Dans le portail de support à la clientèle, cliquer sur **Downloads**, puis **JBoss Operations Network**, puis sélectionner **JBoss ON for BRMS**
- b. Cliquer sur **Download** pour commencer le téléchargement.

2. Décompresser le téléchargement du plug-in.

Décompresser le fichier jar du fichier zip téléchargé :

```
unzip jon-plugin-pack-brms-2.4.1.GA.zip
```

3. Connectez-vous à l'interface utilisateur JBoss ON

Rendez-vous sur la page <http://localhost:7080/Login.do>, et connectez-vous.

4. Télécharger le fichier jar de plug-in.

Sélectionner **Administration**, à partir du menu, puis **System Configuration, Plugins**. Cliquer le bouton **Add** (ajouter) et sélectionner le fichier jar extrait de l'étape précédente, et cliquer sur **upload**.

9.2. DÉCOUVREZ L'INSTANCE BRMS

Après avoir installé le plug-in, **JBoss Operations Network** a besoin de découvrir quelle instance BRMS doit être surveillée.

JBoss Operations Network découvrira BRMS automatiquement. Cependant, si cela ne fonctionne pas, on peut ajouter les ressources manuellement, veuillez consulter *JBoss Operations Network Basic Admin Guide* pour obtenir davantage d'informations.

Après que BRMS a été découvert, il doit être importé dans l'inventaire.

Procédure 9.3. Import de BRMS

1. À partir de la file d'attente Discovery Queue, sélectionner le processus JMX, et cliquer sur **Import**.
2. Sélectionner serveurs à partir du menu **Resources**.
3. Sélectionner le serveur JMX pour visualiser le service BRMS.

NOTE

Les MBeans doivent être activés pour que JBoss ON puisse surveiller BRMS.

Les MBeans peuvent être activés, soit en passant le paramètre :

```
-Ddrools.mbeans = enabled
```

Ou via l' API :

```
KnowledgeBaseConfiguration conf =  
    KnowledgeBaseFactory.newKnowledgeBaseConfiguration();  
conf.setOption( MBeansOption.ENABLED );
```

ANNEXE A. DÉPLOYER JBOSS BRMS SUR EWS

BRMS peut être installé sur JBoss Enterprise Web Server 1.0.1 avec Tomcat 5 et 6.

Pour les instructions d'installation et de configuration de **JBoss Enterprise Web Server**, consulter *JBoss Enterprise Web Server Installation Guide*.

Procédure A.1. Installer le package déployable sur le serveur JBoss Enterprise Web

1. Télécharger BRMS

Télécharger le fichier zip BRMS Deployable Package à partir du portail de support à la clientèle de Red Hat <https://access.redhat.com>. Naviguez dans **Downloads**, puis **BRMS Platform**, enfin, sélectionner la version.

2. Extraire le logiciel

Extraire les fichiers déployables BRMS du fichier zip

```
[localhost]$ unzip brms-deployable-5.2.0.zip
```

Cela va créer les fichiers zip suivants :

- **jboss-brms.engine.zip**
- **jboss-brms.manager.zip**

Extraire **jboss-brms.war** de **jboss-brms-manager.zip**. Par exemple :

```
[localhost]$ unzip brms-manager.zip
```

Cela va créer un répertoire **jboss-brms.war** qui contient l'application web JBoss Enterprise BRMS Platform comme archive explosée.

Extraire les bibliothèques client de **jboss-brms-engine.zip** et les mettre dans un répertoire. Par exemple :

```
[localhost]$ unzip brms-engine.zip -d client
```

Cela crée un répertoire qui contient les fichiers jar de la bibliothèque du client.

3. Copier jboss-brms.war dans le serveur d'applications.

Copier le répertoire extrait **jboss-brms.war** dans le répertoire EWS suivant, **tomcatversion/webapps/jboss-brms/WEB-INF/**, où la *version* correspond au numéro 5 ou 6.

4. Copier les bibliothèques slf4j dans l'app web déployée.

Copier les bibliothèques slf4j libraries à partir du répertoire **client** créé dans l'étape précédente dans l'app web BRMS déployée :

```
$ cp client/lib/slf4j* jboss-brms.war/WEB-INF/lib/
```

5. Activer les utilisateurs

Créer un LoginModule personnalisé pour Tomcat 5 ou 5, veuillez consulter la documentation Tomcat pour obtenir des informations à ce sujet.

6. **Démarrer le serveur**

Naviguer dans **tomcat`version`/bin/** et utiliser la commande **run.sh** sur un système UNIX ou Linux, ou run.bat sur un système Microsoft Windows pour lancer le serveur.

7. **Connexion**

Connectez-vous à l'application web JBoss Enterprise BRMS Platform pour confirmer qu'elle est installée et qu'elle exécute correctement. Voir [Section 2.3](#), « [Première connexion](#) » pour plus de détails.

ANNEXE B. INSTALLER APACHE ANT

L'outil *Apache Ant* de Java et le plugin Trax sont requis pour l'installation de JBoss ModeShape. Non requis pour l'installation ou les opérations normales dans JBoss Enterprise BRMS Platform. Apache Ant requiert un JRE (Java Runtime Environment) correct.

Si vous exécutez un poste de travail de développement, Apache Ant est sans doute déjà installé. Le package Apache Ant du site web Apache Ant inclut le plugin Trax mais Red Hat Enterprise Linux l'inclut comme package séparé.

Procédure B.1. Installer Apache Ant sur Red Hat Enterprise Linux

- Télécharger et installer Apache Ant et le plugin Trax sur le référentiel de Red Hat Enterprise Linux à l'aide de cette commande :

```
[localhost]$ sudo yum install ant-trax
```

Procédure B.2. Installer Apache Ant sur les autres systèmes d'exploitation

1. Téléchargement et Extraction

Télécharger la version binaire de **Apache Ant** à partir de <http://ant.apache.org/bindownload.cgi>.

Une fois téléchargée, l'extraire dans une location d'installation qui vous convient, comme **c:\Program Files\Apache\Ant** ou **/opt/apache-ant-1.8/**.

2. Ajouter la variable d'environnement **ANT_HOME**

Créer une variable d'environnement intitulée **ANT_HOME**. Cette variable doit contenir le chemin créé dans l'étape précédente.

- Procédez sur Red Hat Enterprise Linux en ajoutant la ligne suivante au fichier **~/.bash_profile**, en remplaçant le chemin par celui qui a été créé dans l'étape précédente.

```
export ANT_HOME=/opt/apache-ant-1.8.1
```

- Dans Microsoft Windows, procédez en cliquant sur **Start Menu**, en ouvrant **Control Panel** puis en sélectionnant **System -> Advanced -> Environment Variables**.

Créer une nouvelle variable, en l'appelant **ANT_HOME** et configurez-la pour qu'elle pointe vers le répertoire créé dans l'étape précédente.

3. Inclure bin dans le PATH

Ajouter le répertoire **bin** de l'installation Ant à la variable d'environnement **PATH**.

- Sur les systèmes Unix/Linux, on procède tout simplement en ajoutant la ligne suivante au fichier **~/.bash_profile** après celle qui définit la variable **ANT_HOME** :

```
export PATH=$PATH:$ANT_HOME/bin
```

- Dans Microsoft Windows, procédez en ouvrant le **Control Panel** et en sélectionnant **System -> Advanced -> Environment Variables -> System Variables**. Modifier la variable **PATH** et ajouter le texte **;%ANT_HOME%\bin**.

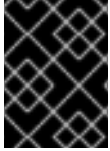
Pour tester l'installation Apache Ant, exécuter **ant -version** à partir d'une console shell manuelle. La sortie devra ressembler à ceci :

```
[localhost]$ ant -version  
Apache Ant version 1.8 compiled on June 27 2008
```

Pour en savoir davantage sur **Apache Ant**, consulter le site web du projet à <http://ant.apache.org>.

ANNEXE C. EXEMPLE DE CONFIGURATION DU GESTIONNAIRE DE PERSISTANCE

Il existe plusieurs exemples de configuration du gestionnaire de persistance pour les bases de données prises en charge. Vous devrez, bien sûr, mettre à jour les valeurs dans les configurations avec celles qui sont correctes dans votre base de données, comme URL JDBC et schemaObjectPrefix.



IMPORTANT

JBoss Enterprise BRMS Platform est supporté sur les bases de données listées ici :
<http://www.jboss.com/products/platforms/brms/supportedconfigurations/>

Pour obtenir des informations sur les gestionnaires Apache Jackrabbit Persistence Managers, vous devrez consulter <http://wiki.apache.org/jackrabbit/PersistenceManagerFAQ>

Exemple C.1. Configuration générique JDBC pour MySQL avec BundleDbPersistenceManager

```
<PersistenceManager class=
"org.apache.jackrabbit.core.persistence.bundle.BundleDbPersistenceManager">
  <param name="driver" value="com.mysql.jdbc.Driver"/>
  <param name="url" value="jdbc:mysql://localhost/brms"/>
  <param name="user" value="brms_user"/>
  <param name="password" value="brms_password"/>
  <param name="schema" value="mysql"/>
  <param name="schemaObjectPrefix" value="${wsp.name}_"/>
</PersistenceManager>
```

Exemple C.2. Configuration MySQL avec MySqlPersistenceManager

```
<PersistenceManager class=
"org.apache.jackrabbit.core.persistence.bundle.MySqlPersistenceManager">
  <param name="driver" value="com.mysql.jdbc.Driver"/>
  <param name="url" value="jdbc:mysql://localhost:3306/brms"/>
  <param name="user" value="brms_user"/>
  <param name="password" value="brms_password"/>
  <param name="schemaObjectPrefix" value="${wsp.name}_"/>
  <param name="schema" value="mysql"/>
</PersistenceManager>
```

Exemple C.3. Configuration Oracle avec OraclePersistenceManager

```
<PersistenceManager class=
"org.apache.jackrabbit.core.persistence.bundle.OraclePersistenceManager">
  <param name="driver" value="oracle.jdbc.OracleDriver"/>
  <param name="url" value="jdbc:oracle:thin:@localhost:1521:brms" />
  <param name="schema" value="oracle"/>
```

```
<param name="user" value="brms_user" />
<param name="password" value="brms_password" />
<param name="schemaObjectPrefix" value="${wsp.name}_" />
</PersistenceManager>
```

Exemple C.4. Configuration PostgreSQL avec PostgreSQLPersistenceManager

```
<PersistenceManager
class="org.apache.jackrabbit.core.persistence.bundle.
PostgreSQLPersistenceManager">
  <param name="driver" value="org.postgresql.Driver"/>
  <param name="url" value="jdbc:postgresql://localhost:5432/brms" />
  <param name="schema" value="postgresql"/>
  <param name="user" value="brms_user" />
  <param name="password" value="brms_password" />
  <param name="schemaObjectPrefix" value="${wsp.name}_" />
</PersistenceManager>
```

Exemple C.5. Configuration Microsoft SQL Server 2005 avec MSSqlPersistenceManager

```
<PersistenceManager
class="org.apache.jackrabbit.core.persistence.bundle.
MSSqlPersistenceManager">
  <param name="driver"
    value="com.microsoft.sqlserver.jdbc.SQLServerDriver"/>
  <param name="url"
    value="jdbc:sqlserver://localhost:3918;DatabaseName=brms" />
  <param name="user" value="brms_user" />
  <param name="password" value="brms_password" />
  <param name="schema" value="mssql"/>
  <param name="schemaObjectPrefix" value="${wsp.name}_" />
</PersistenceManager>
```

ANNEXE D. HISTORIQUE DE RÉVISION

Version 5.2.0-2.400 Rebuild with publican 4.0.0	2013-10-31	Rüdiger Landmann
Version 5.2.0-2 Rebuild for Publican 3.0	2012-07-18	Anthony Towns
Version 5.2.0-2 Mise à jour pour 5.2.0 Section Surveillance ajoutée Instructions d'installations EWS ajoutées	Tue Jul 1 2011	L Carlon
Version 5.1.0-1 Instructions ModeShape ajoutées	Tue Mar 1 2011	Darrin Mison
Version 5.1.0-0 Créé pour 5.1.0	Tue Aug 17 2010	David Le Sage, Darrin Mison