

1 Installation de base Serveur JBoss

Le serveur d'applications JEE JBoss est accessible à l'adresse web suivante :

<http://www.jboss.org/jbossas/downloads>.

La version 7.1.1 de ce serveur est également disponible dans le répertoire suivant :

`~/ocaron/public/AL/serveurJBoss/jboss-as-7.1.1.Final.tar.gz`

Important : le serveur JBoss version 7 nécessite un interpréteur Java version 6 ou supérieur.

1.1 Installation

L'installation de base nécessite les deux opérations suivantes :

1. Extraction du fichier `jboss-as-7.1.1.Final.tar.gz` (ligne 5 de l'exemple)
2. Spécification d'un administrateur à l'aide de la commande `add-user.sh` (lignes 7-20 de l'exemple)

Voici un exemple d'installation du serveur JBoss 7 avec la définition d'un administrateur de nom `jeeadm` et de mot de passe "`secret`"

```
1 %> java -version
2 java version "1.6.0_17"
3 Java(TM) SE Runtime Environment (build 1.6.0_17-b04)
4 Java HotSpot(TM) Server VM (build 14.3-b01, mixed mode)
5 %> tar xvfz ~/ocaron/public/AL/serveurJBoss/jboss-as-7.1.1.Final.tar.gz
6 %> cd jboss-as-7.jboss-as-7.1.1.Final/bin
7 %> ./add-user.sh
8
9 What type of user do you wish to add?
10 a) Management User (mgmt-users.properties)
11 b) Application User (application-users.properties)
12 (a):
13
14 Enter the details of the new user to add.
15 Realm (ManagementRealm) :
16 Username : jeeadm
17 Password :
18 Re-enter Password :
19 About to add user 'jeeadm' for realm 'ManagementRealm'
20 Is this correct yes/no? yes
21 ...
```

Pour la suite, nous considérons que la variable d'environnement `${JBOSS_HOME}` désigne le répertoire du serveur. Idéalement, votre fichier `.bashrc` devrait contenir les deux lignes suivantes :

```
export JBOSS_HOME=/home/.../jboss-as-7.jboss-as-7.1.1.Final
export PATH=${PATH}:${JBOSS_HOME}/bin
```

1.2 Lancement du serveur

La version 7 du serveur JBoss se caractérise par un changement rapide (une dizaine de secondes) car la plupart des services ne sont effectivement chargés en mémoire que lorsqu'ils sont sollicités. La commande pour lancer le serveur est la suivante :

```
%> cd ${JBOSS_HOME}/bin
%> ./standalone.sh
...
7.1.1.1.Final "Brontes" started in 3173ms - Started 133 of 208 services (74 services
are passive or on-demand)
```

Pour arrêter ce serveur, un simple CTRL-C suffit.

Lorsque le serveur JEE est actif, une console web d'administration permet de superviser l'activité du serveur et est disponible à l'adresse `http://localhost:8080`, sélectionnez ensuite "Administration Console", un login/password vous est demandé, indiquez les nom et mot de passe de l'administrateur que vous avez créé préalablement avec la commande `add-user.sh`.

2 Ajout d'une source de données

L'objectif est de compléter l'installation de base avec l'ajout d'une nouvelle source de données qui sera une base de données d'un serveur Postgres. Cette source sera accessible par le serveur de nom JNDI inclus dans le serveur JEE. Cette installation se fait en deux étapes :

- Installation du driver JDBC dédié au serveur postgres
- Installation de la source de données.

2.1 Driver Postgres

L'architecture du serveur JBoss est modulaire. Toute librairie Java est déployée dans l'arborescence à partir de `${JBOSS_HOME}/modules` avec un descripteur XML qui indique les dépendances du module déployé avec d'autres librairies. La procédure d'installation est la suivante :

1. Création du répertoire où sera déployé le module (lignes 2-3 de l'exemple)
2. Stockage du driver (ligne 5 de l'exemple). Le driver se trouve également dans le répertoire : `~ocaron/public/AL/serveurJBoss/`
3. Stockage du descripteur XML (ligne 6 de l'exemple) :

```
<module xmlns="urn:jboss:module:1.1" name="org.postgresql">
  <resources>
    <resource-root path="postgresql.jar" />
  </resources>
  <dependencies>
    <module name="javax.api" />
    <module name="javax.transaction.api" />
  </dependencies>
</module>
```
4. Activation du module (lignes 7-10) de l'exemple. Le serveur JBoss doit être lancé lors de l'activation du module.

Voici un exemple d'installation de ce module :

```
1 %> cd ${JBOSS_HOME}/modules
2 %> mkdir org/postgresql
3 %> mkdir org/postgresql/main
4 %> cd org/postgresql/main
5 %> cp /usr/share/java/postgresql.jar .
6 %> cp ~ocaron/public/AL/serveurJBoss/module.xml .
7 %> ${JBOSS_HOME}/bin/jboss-cli.sh --connect command="/subsystem=datasources/
8 jdbc-driver=postgresql-driver:add(driver-name=postgresql-driver,
9 driver-class-name=org.postgresql.Driver, ,driver-module-name=org.postgresql"
10 {"outcome" => "success"}
11 %>
```

Pour éviter des erreurs de saisie, le script `loadDriverPostgres.sh` qui exécute ces commandes est disponible dans le répertoire : `~ocaron/public/AL/serveurJBoss`

2.2 Définition de la source de données

Voici la procédure pour créer et définir une nouvelle source de données :

- Création d'une base de données de nom `jeedbVotreLogin` sur le serveur POSTGRES de la machine `weppes.studserv.deule.net` de l'école (rappel commande `createdb jeedbVotreLogin`).
- Enregistrement de cette base dans le serveur JBoss via la console d'administration web :
 1. Accédez à `http://localhost:8080`
 2. Cliquez sur `Administration Console`, rentrez login et mot de passe de l'administrateur.
 3. Sélectionnez l'onglet "`Profile`" (en haut, à droite). Dans la partie "`Datasources`", cliquez sur le bouton "`Add`".
 4. Saisissez "`PostgresDS`" et "`java:jboss/datasources/PostgresDS`" pour les champs "`Name`" et "`JNDI Name`", cliquez sur "`Next`".
 5. Sélectionnez "`postgresql-driver`", cliquez sur "`Next`".
 6. Saisissez "`jdbc:postgresql://weppes.studserv.deule.net/jeedbVotreLogin`", votre nom de login, et votre mot de passe pour les champs "`Connection URL`", "`Username`", et "`Password`". Les noms de login et password demandés correspondent à votre compte enregistré par le serveur Postgres. Par défaut, le login du compte postgres est le même que le login UNIX (les caractères '-' sont remplacés par '_') et le mot de passe par défaut est "`postgres`". Laissez le champ "`Security Domain`" vide puis cliquez sur "`Done`".
 7. Sélectionnez votre nouvelle source de données et cliquez sur le bouton "`enable`".

Cette source de données sera utilisée lors de la programmation de composants EJB entités.