Secrétariat général
de la défense et de la sécurité nationale

Agence nationale de la sécurité
des systèmes d'information

Paris, le 13 décembre 2012

N° ANSSI/SIS/BSH/NO-REFERENCE

Nombre de pages du document : 9

Environnement de développement

Installation

DIFFUSION RESTREINTE
Ce document ne doit être communiqué qu'aux personnes qualifiées pour le connaître.

Document version 1.0.0

51 boulevard de La Tour-Maubourg 75700 PARIS 07 SP - Tél 01.71.75.84.65 - Fax 01.71.75.84.90

Table des matières

Liste des scripts

1 Objectif

L'objectif de ce document est de décrire la procédure d'installation des éléments qui composent la chaine de dévelopement utilisée à l'ANSSI. Cette procédure décrira également comment importer des données pré-existantes dans ces éléments. Cette version du document décrit l'installation des éléments suivants :

- Gerrit Code Review
- Jenkins CI
- Redmine

Ce document assume que vous procédez à l'installation de ces éléments sur Ubuntu 12.04 64bits.

Gerrit et Jenkins font tourner leur propre serveur Web, ils seront donc placés derrière un proxy Apache. Redmine est lui servi par Apache et son `mod_passenger`. Il peut également être placé derrière un proxy web.

2 Gerrit Code Review

2.1 Gerrit, adapté pour l'ANSSI

La version de Gerrit utilisée par l'ANSSI a été adaptée pour ses propres besoins. Les modifications restent légères et ne changent en rien la procédure d'installation par rapport à une version officielle de Gerrit. Cependant, lors d'une mise à jour de Gerrit, il faut prévoir de répercuter ces modifications.

Cette opération ne devrait pas poser de problème pour les versions 2.5.x de Gerrit. Rien n'est garanti pour des versions supérieures à la 2.5, mais il est cependant recommandé de répercuter les modifications à chaque nouvelle version de Gerrit afin de rester au plus près du code officiel.

La version de Gerrit adapté à l'ANSSI est numérotée comme suit : *a.b.c.d* avec *a.b.c* la version officielle de Gerrit sur laquelle est basée cette version de Gerrit et *d* le numéro de version interne à l'ANSSI.

Par exemple, si la version actuelle de Gerrit est la 2.5.0, et qu'il y a eu 5 versions des modifications internes à l'ANSSI, alors le numéro sera 2.5.0.5. Si Gerrit 2.5.1 sort et que les modifications sont répercutées dessus alors, le numéro sera 2.5.1.5 (et pas 2.5.1.0). Autrement dit, le "d" est totalement indépendant du "a.b.c".

La version utilisée pour la rédaction de ce document est la 2.5.0.5.

2.2 Données à récupérer d'une ancienne installation

Si vous voulez récupérer les données d'une ancienne installation de Gerrit, voici ce que vous devriez récupérer :

- Un dump de la base de données
- Les projets Git
- Les fichiers de configuration

Dump de la base de données

```
operateur@oldserver:~$ sudo su postgres
postgres@oldserver:~$ pg_dump reviewdb > /tmp/reviewdb.sql
```

`reviewdb` est le nom de la base de données utilisée par Gerrit. Si ce nom a changé, adaptez les commandes en conséquence.

Archivage des projets Git

```
operateur@oldserver:~$ cd /var/gerrit/projects
operateur@oldserver:~$ tar czvf gerrit-projects.tgz *
```

Le chemin `/var/gerrit/projects` est celui par défaut, mais il se peut aussi que les projets se trouvent dans `/home/gerrit`.

Attention, le fichier `gerrit-projects.tgz` peut être volumineux (12Go lors des tests). Il se peut que vous deviez le découper en plusieurs parties pour pouvoir le transporter. Dans ce cas, utilisez les commandes suivantes :

Archive découpée des projets Git

```
operateur@oldserver:~$ cd /var/gerrit/projects
operateur@oldserver:/var/gerrit/projects$ tar czv * | split -b
    1024MiB - gerrit-projects.tgz_
```

Archivage des fichiers de configuration

```
operateur@oldserver:~$ cd /srv/gerrit
operateur@oldserver:/srv/gerrit$ tar czvf gerrit-etc.tgz etc/
```

Le dossier d'installation de Gerrit est par défaut `/srv/gerrit`, mais il se peut aussi qu'il soit `/opt/gerrit`. Le dossier `<gerrit-install-dir>/etc` contient tous les fichiers de configuration dont vous pourriez avoir besoin.

Vous pouvez également avoir besoin de récupérer la configuration Apache pour restaurer un proxy web identique. Dans ce cas, le fichier que vous devez sauvegarder se trouve évidemment sur la machine qui sert de proxy web et s'appelle `/etc/apache2/sites-available/gerrit-proxy`.

2.3 Installation de Gerrit

Avant d'installer Gerrit, assurez-vous que le JRE, PostgreSQL, Apache (si vous installez le proxy web directement sur la machine) et Gitweb soient installés sur la machine. Tous sont disponibles à travers le gestionnaire de paquet.

Les fichiers `reviewdb.sql`, `gerrit-etc.tgz` et `gerrit-projects` ont été copiés dans `/home/operateur` et sont lisibles par tous les utilisateurs de la machine. Le fichier `.war` de Gerrit se trouve au même endroit, il est lisible par tous et se nomme `gerrit-2.5.0.5-ANSSI.war`.

Installation de Gerrit – Configuration

```
operateur@newserver:~$ sudo adduser --no-create-home gerrit
operateur@newserver:~$ sudo mkdir -p /srv/gerrit/etc
operateur@newserver:~$ cd /srv/gerrit/etc
operateur@newserver:/srv/gerrit/etc$ sudo tar xvf /home/operateur/
   gerrit-etc.tgz
operateur@newserver:/srv/gerrit/etc$ cd
operateur@newserver:~$ sudo chown -R gerrit:gerrit /srv/gerrit
```

Installation de Gerrit – Projets Git

```
operateur@newserver:~$ sudo mkdir -p /var/gerrit/projects
operateur@newserver:~$ sudo chown -R gerrit:gerrit /var/gerrit/
   projects
operateur@newserver:~$ sudo su gerrit
gerrit@newserver:/home/operateur$ cd /var/gerrit/projects
gerrit@newserver:/var/gerrit/projects$ tar xvf /home/operateur/
   gerrit-projects.tgz
```

Vous pouvez utiliser la commande `cat /home/operateur/gerrit-projects.tgz_* | tar xvf` à la place de la dernière ligne si l'archive contenant les projets a été découpée.

Installation de Gerrit – Base de données

```
operateur@newserver:~$ sudo su postgres
postgres@newserver:/home/operateur$ createuser -D -P -E gerrit
postgres@newserver:/home/operateur$ createdb -E UTF-8 -O gerrit
   reviewdb
postgres@newserver:/home/operateur$ psql -d reviewdb
reviewdb=# \i /home/operateur/reviewdb.sql
```

Lors de la création de l'utilisateur PostgreSQL `gerrit` (deuxième ligne), répondez « non » aux deux questions : `gerrit` n'est pas un *superuser* et il ne peut pas créer de nouveaux rôles.

Installation de Gerrit

```
1 operateur@newserver:~$ sudo su gerrit
2 gerrit@newserver:/home/operateur$ java -jar gerrit-2.5.0.5-ANSSI.
    war init -d /srv/gerrit
3 gerrit@newserver:/home/operateur$ exit
4 operateur@newserveur:~$ sudo ln -snf /srv/gerrit/bin/gerrit.sh /
    etc/init.d/gerrit
5 operateur@newserveur:~$ sudo update-rc.d gerrit defaults
6 operateur@newserveur:~$ sudo echo "GERRIT_SITE=/srv/gerrit" > /etc
    /default/gerritcodereview
7 operateur@newserver:~$ sudo su gerrit
8 gerrit@newserver:/home/operateur$ service gerrit restart
```

La ligne 2 installe Gerrit dans le dossier `/srv/gerrit`. Comme ce dernier contient déjà des fichiers de configuration, l'installeur vous proposera les valeurs contenues dans ces fichiers comme valeurs par défaut. Adaptez les mots de passe à votre nouvelle installation.

Toujours lors de l'exécution de la ligne 2, indiquez bien `/var/gerrit/projects` comme emplacement des projets au début de l'installation.

Pensez quand même à vérifier que ces valeurs sont les bonnes, notamment le chemin vers l'exécutable `gitweb.cgi` et les adresses sur lesquelles Gerrit doit écouter (ces informations sont dans le fichier `/srv/gerrit/etc/gerrit.config`).

Les lignes 4 et 5 permettent de faire de Gerrit un service qui démarrera automatiquement à l'allumage du serveur et s'éteindra à l'extinction.

La ligne 6 est importante, sans elle Gerrit ne pourra pas démarrer.

2.4 Proxy Apache pour Gerrit

Le proxy web peut être installé sur la même machine que Gerrit ou sur une machine distante.

Dans le premier cas, pensez à indiquer `127.0.0.1` comme adresse d'écoute pour l'application web lors de l'installation. Dans le second cas, utilisez l'adresse de l'interface qui communiquera avec le proxy web. Si vous avez mal configuré cette partie lors de l'installation, vous pouvez modifier le fichier `/srv/gerrit/etc/gerrit.config` puis redémarrer l'application avec `service gerrit restart` (les deux opérations étant faites avec l'utilisateur `gerrit`).

Sur la machine qui fait office de proxy web, installez Apache2 et activez le `mod_proxy_http` :

Initialisation d'Apache

```
operateur@proxyweb:~$ sudo apt-get install apache2
operateur@proxyweb:~$ sudo a2enmod proxy_http
```

Puis créez un fichier `/etc/apache2/sites-available/gerrit-proxy` qui contient ce qui suit :

VHOST proxy Gerrit

```
<VirtualHost *:80>
    ServerName <server-name>
    ErrorLog "/var/log/apache2/gerrit-error_log"
    CustomLog "/var/log/apache2/gerrit-access_log" common

    ProxyRequests Off
    ProxyVia Off
    ProxyPreserveHost On
    ProxyErrorOverride On

    <Proxy *>
        Order deny,allow
        Allow from all
    </Proxy>

    ProxyPass / http://<gerrit-server-ip>:8080/
    ProxyPassReverse / http://<gerrit-server-ip>:8080/
</VirtualHost>
```

Remplacez `<server-name>` et `<gerrit-server-ip>` par les valeurs correspondantes. Vous avez normalement déjà indiqué le `<server-name>` lors de l'installation de Gerrit [1]. Adaptez la configuration du VHOST en fonction de votre configuration actuelle. Si vous avez sauvegardé le précédent VHOST, vous pouvez bien sûr réutiliser celui-ci.

Et enfin, activez le site et redémarrez Apache :

Fin de la configuration d'Apache

```
operateur@proxyweb:~$ sudo a2ensite gerrit-proxy
operateur@proxyweb:~$ sudo service apache restart
```

2.5 Compte pour l'utilisateur Jenkins

Jenkins doit pouvoir se connecter à Gerrit, et doit donc avoir un compte correspondant dans le serveur LDAP.

Si avez fait une installation propre de Gerrit (sans import de base de données) Connectez-vous une première fois en tant que Jenkins sur Gerrit pour créer le compte, puis connectez-vous en tant qu'administrateur et ajoutez Jenkisn au groupe « Non-Interactive Users ». Après l'installation de Jenkins, vous devrez vous reconnecter à Gerrit pour indiquer sa clé SSH.

1. Si besoin, vous le trouverez dans le fichier `/srv/gerrit/etc/gerrit.config`, sous le nom `gerrit.canonicalWebUrl`.

3 Jenkins Continuous Integration

La version de Jenkins actuellement utilisée est la 1.466.2 : c'est la dernière version LTS qui a été packagée pour Ubuntu. L'équipe de développement de Jenkins sort un grand nombre de versions avec assez peu de différences entre chacune. Pour éviter des mises à jour trop fréquentes, l'ANSSI n'utilise que les versions LTS pré-packagées pour Ubuntu.

3.1 Plugins utilisés à l'ANSSI

Les plugins suivants sont utilisés par l'ANSSI. Ils seront installés automatiquement par l'installeur fourni, ils ne sont donc donnés qu'à titre indicatif :

- Gerrit Trigger : intégration avec Gerrit, automatisation du lancement d'une compilation à chaque nouveau *changeset* sur Gerrit.
- Git : permet à Jenkins d'utiliser Git comme système de gestion de versions.
- Ant : permet à Jenkins d'utiliser Ant.
- LDAP : permet à Jenkins d'utiliser un annuaire LDAP pour l'authentification.
- Repo : permet à Jenkins d'utiliser Repo (outil qui permet de travailler sur plusieurs dépôts Git à la fois).
- SSH-slaves : contrôle d'esclaves par SSH pour la compilation.
- Subversion : permet à Jenkisn d'utiliser SVN comme système de gestion de versions.

3.2 Données à récupérer d'une ancienne installation

Les fichiers que vous devez sauvegarder se trouvent dans le dossier `/var/lib/jenkins`. Le plus simple est de sauvegarder l'intégralité du dossier en excluant éventuellement les dossiers du type `jobs/<nom-du-job>/builds` ou encore les dossiers `jobs/<nom-du-job>/workspace` :

Sauvegarde des données de Jenkins

```
operateur@oldserver:~$ sudo tar czvf jenkins.tgz --exclude='jobs
   /*/builds' --exclude='jobs/*/builds/*' --exclude='jobs/*/
   workspace' --exclude='jobs/*/workspace/*' /var/lib/jenkins
```

Si vous y avez apporté des modifications, n'oubliez pas de sauvegarder les deux fichiers suivants :

- `/etc/default/jenkins`
- `/etc/apache2/sites-available/jenkins`

Enfin, bien que présents dans l'archive `jenkins.tgz` créée précédemment, il est plus aisé de sauvegarder à part les fichiers de configurations des jobs Jenkins. Pour cela, vous pouvez utiliser le script suivant (adaptez-le si besoin) :

Sauvegarde des jobs Jenkins

```
#!/bin/sh
INSTALL_PATH=/var/lib/jenkins
JOBS_PATH=$INSTALL_PATH/jobs
BKP_DIR=jobs-jenkins
mkdir -p $BKP_DIR
for J in $JOBS_PATH/*; do
        NAME=$(echo $J | sed 's/\/.*\///g' | sed 's/ //g')
        echo "Backuping config of job $NAME"
        cp "$J/config.xml" "$BKP_DIR/$NAME.xml"
done
tar czvf jobs-jenkins.tgz $BKP_DIR
rm -r $BKP_DIR
```

Ce script crée une archive qui contient le fichier de configuration de chaque job. Chaque fichier est nommé `<nom-du-job>.xml`.

3.3 Installation de Jenkins

Si vous voulez récupérer des données précédemment sauvegardées, extrayez les archives créées en section 3.2 ainsi que l'installeur :

Jenkins - Désarchivage

```
operateur@newserver:~$ tar xvf jenkins.tgz
operateur@newserver:~$ tar xvf jobs-jenkins.tgz
operateur@newserver:~$ tar xvf jenkins-installer-v1.1.0.tgz
```

Puis lancez l'installation en tant que telle :

Jenkins - Installation

```
operateur@newserver:~$ cd jenkins-installer-v1.1.0
operateur@newserver:~$ sudo ./install.sh
```

Suivez les instructions du script. Quand il vous demande si vous souhaitez importer des jobs (et si c'est le cas), indiquez lui le chemin vers `/home/operateur/jobs-jenkins` (ou l'endroit où vous avez désarchivé `jobs-jenkins.tgz`). Si vous voulez importer une ancienne configuration, indiquez-lui le chemin vers `/home/operateur/var/lib/jenkins` (ou l'endroit où vous avez

désarchivé `jenkins.tgz`).

Clé SSH
À la fin de l'installation, une clé publique SSH est affichée. Cette clé se trouve dans le fichier `/var/lib/jenkins/.ssh/id_rsa.pub`. Vous devez ajouter cette clé comme clé publique à l'utitlisateur `jenkins` sur Gerrit (connectez-vous à Gerrit en tant que Jenkins, puis dirigez-vous vers *Settings* > *SSH Public Keys*). Elle serviera également pour la connexion aux esclaves.

Clé SSH - Password
Pour l'instant il n'est pas possible de mettre un mot de passe à cette clé : le plugin Gerrit Trigger semble buggué à ce niveau là. La clé est donc créée sans mot de passe.

Je ne peux pas me connecter à l'interface web !
Il est probable que la configuration de l'ancien serveur ne soit pas totalement adapté au nouveau, notamment au niveau de l'authentification. Vous pouvez modifier manuellement cette configuration en éditant le fichier `/var/lib/jenkins/config.xml` puis en redémarrant Jenkins avec la commande `sudo service jenkins restart`.

Mauvaise détection de la version de Gerrit
Jenkins indique que la version détectée de Gerrit est ancienne et que toutes les fonctionnalités n'ont pas été activées. Cela est dû au fait que la version de Gerrit utilisée est modifiée. Vous pouvez ignorer cet avertissement pour le moment.

Il est relativement probable que les configurations que vous importez ne fonctionnent pas du premier coup et demandent quelques modifications. Faites en particulier attention à :

- La gestion des esclaves (voir section 3.4)
- La configuration de Gerrit Trigger

3.4 Gestion des esclaves

Si vous devez réinstaller les esclaves de compilation, utilisez Debian Wheezy 64bits comme distribution. En effet, dans l'optique du futur déploiement de Coverity®, il est nécessaire d'utiliser cette distribution car ce logiciel ne fonctionne pas sur Ubuntu 12.04.

Dans tous les cas, vérifiez que l'utilisateur `jenkins` du nouveau serveur a un accès SSH aux esclaves sans mot de passe (avec sa clé SSH).

Si vous devez configurer un nouvel esclave dans Jenkins, assurez-vous qu'il a bien accès à cette nouvelle machine et rendez-vous dans la section « Gérer les nœuds » du tableau d'administration de Jenkins. Cliquez sur « Nouveau nœud », rentrez un nom, choisissez « Esclave passif » et dans l'écran suivant, choisissez « Launch slave agents on Unix machines via SSH » comme méthode de lancement [2] puis configurez l'esclave comme vous le souhaitez.

2. Grâce à cette option, Jenkins va se connecter en SSH à l'esclave et y copier tous les exécutables dont il a besoin.

Logiciels nécéssaires sur les esclaves

Vous devez installer les compilateurs que vous comptez installer sur les esclaves : GCC, *javac* (via *openjdk*)... Ainsi que Git et Repo. Tous ces logiciels doivent être dans le `$PATH` de l'utilisateur que Jenkins utilise sur cet esclave.

- FIN DU DOCUMENT -