

Mise en place d'une réplique de serveur par Rsync via SSH

I. Principes

Nous allons baser ce document sur l'exemple de deux serveurs http1 et http2, http1 est le serveur web principal et http2 doit garder une configuration identique pour remplacer http1 au cas ou. Le principe de base est le suivant :

- La machine http2 se connecte sur la machine http1 via ssh
- Rsync est exécuter sur la machine http1, celui-ci applique un algorithme de différenciation et ne transfère que les fichiers qui diffèrent. Avec l'option `-delete`, il supprime les fichiers en trop sur http2 de sorte que la réplique est parfaite.
- Une fois la réplique des répertoires effectuée un mail est envoyé à root contenant le détail des opérations.
- Le tout est répété chaque nuit (ou avec un intervalle plus court).

II. Pré-requis

Pour la connexion ssh :

- Ssh doit être installé sur http1 et http2, http1 doit avoir le daemon sshd en exécution (pour le vérifier : `ps -ef | grep sshd`).
- Attention si un firewall local est installé sur http1, il empêche toute connexion sur le port 22 (accès ssh), dans ce cas rajouter la règle suivante :

```
#autorisation de la réplique http2
${IPCHAINS} -A input -j ACCEPT -p TCP -s
<http2_IP_address>/<subnet mask> -d ${IPETH0} 22
```

- Http2 n'utilise ssh qu'en client, il faut générer une paire de clé (publique/privé) avec une phrase secrète : taper `ssh-keygen` puis valider par défaut les trois questions (fichier identity, phrase secrète et phrase vérification).
- La clé publique de http2 se trouve donc dans `/root/.ssh/identity.pub`, éditer ce fichier puis copier la clé.

- Maintenant sur http1, dans /root/.ssh, se trouve le fichier authorized_hosts qui contient les clés publiques des serveurs ayant l'autorisation de se connecter sur http1. Rajouter la clé de http2 à la fin du fichier. Attention, en général le copier coller entre terminal scinde l'unique ligne de départ en plusieurs lignes, penser donc à joindre les lignes (shift J dans VI).
- Pour tester la connexion, placer vous sur http2 et taper ssh <http1_IP_adress>, normalement http1 ne fait pas parti des known_hosts de http 2, il vous demande donc si vous êtes sûr de vouloir vous connecter. Répondre yes : http1 est rajouté au known_hosts de http2 et la question ne sera jamais reposée.

Pour la réplication via Rsync :

- Rsync doit être installer sur http1, il n'est pas nécessaire qu'il soit installé sur http2.

III. Mise en place

Voici un script exemple qui réplique la structure web et ftp de la machine http1 sur http2, bien sur on peut répliquer n'importe quoi d'autre, ici le chroot ftp se trouve dans /u/ftp/users et la structure web dans /u/httpd/services :

```
#!/bin/sh
```

```
rsync=/usr/local/bin/rsync
```

```
{
# Mise à jour des fichiers de conf Apache
${rsync} -av --delete -e /usr/local/bin/ssh http1:/etc/httpd/conf /etc/httpd
cat /etc/httpd/conf/httpd.conf | sed "s|<IP HTTP1>|<IP HTTP2|" > /etc/httpd/conf/httpd.conf.1
cat /etc/httpd/conf/httpd.conf.1 | sed "s|<nom de la machine http1>|<nom de la machine
http2>|" > /etc/httpd/conf/httpd.conf
rm /etc/httpd/conf/httpd.conf.1
```

```
# Redemarrage du serveur Apache après mise à jour des fichiers de conf
/etc/rc.d/init.d/httpd stop
/etc/rc.d/init.d/httpd start
```

```
# Mise à jour du répertoire /u/httpd/services et du répertoire /u/ftp/users
${rsync} -av --delete -e /usr/local/bin/ssh http1:/u/httpd/services /u/httpd
${rsync} -av --delete -e /usr/local/bin/ssh http1:/u/ftp/users /u/ftp
} > /u/tmp/replication 2>&1
```

```
cat /u/tmp/replication | mail -s "Replication des serveurs Web et ftp" root
```

Regardons attentivement ce script, il définit tout d'abord l'emplacement de rsync sur http1 : `rsync=/usr/bin/rsync`

Puis il lance la réplication des répertoires, le choix de ces répertoires doit être judicieux : en effet, la réplication a pour but principal d'avoir une machine backup, il faut donc répliquer tous les répertoires indispensables au bon fonctionnement du serveur.

Pour un serveur web, il s'agit en général du répertoire qui contient la configuration d'apache, qui doit être adaptée à la nouvelle machine, il faut donc appliquer les filtres nécessaires au `httpd.conf`. Ici on remplace dans le `httpd.conf` l'ip de http1 par l'ip de http2 et le nom de la machine http1 par le nom de la machine http2. Bien sur suivant votre configuration d'autres filtres peuvent être nécessaires.

Puis on réplique les répertoires qui contiennent la structure web et ftp. Attention, Rsync ne suit pas les liens symboliques, il faut donc prévoir la réplication des répertoires pointés par des liens.

Pour chaque répertoire à répliquer insérer une ligne de la forme:

```
`${rsync} -av --delete -e /usr/local/bin/ssh http1:/u/httpd/services /u/httpd
```

Explications: rsync est lancé sur la machine http1 via une connexion ssh (bien préciser l'emplacement de ssh sur http2, ici `/usr/local/bin/ssh`) et effectue la réplication du répertoire `services`.

Attention, `/u/httpd/services` est différent de `u/httpd/services/`, en effet dans le premier cas le répertoire `services` est recopié dans `/u/httpd` de http2 tandis que dans le second cas le contenu du répertoire `services` est copié dans `/u/httpd` de http2.

Les messages sont ensuite écrits dans le fichier `/u/tmp/replication`, qui est lui-même envoyé par mail à root.

Pour automatiser cette réplication chaque nuit :

```
Crontab -l > toto.fic
```

```
Vi toto.fic
```

```
Rajouter : 00 03 * * * /usr/local/bin/replication pour une réplication à 3h du matin
```

```
Crontab toto.fic
```

```
Rm toto.fic
```

Les deux machines ont donc maintenant la même configuration web, la même structure et peuvent donc être interchangeables.