

Haute disponibilité avec HeartBeat

Emetteur(s) : Matthieu SAVIARD

Destinataire(s) : Jury BTS SIO

Direction XX

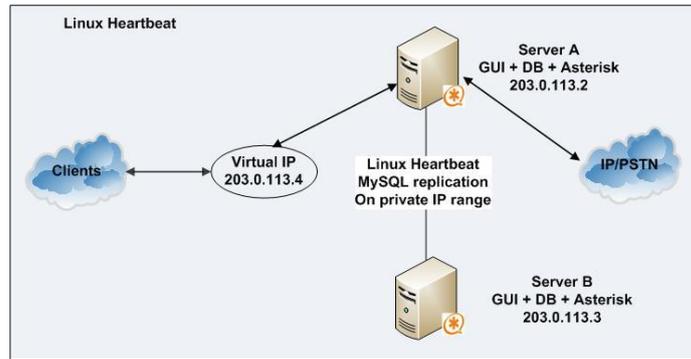
Date : 04/12/2023

Objet : Mise en production d'un service haute disponibilité HeartBeat sur deux serveurs apache

1. Introduction

HeartBeat constitue un logiciel de surveillance de la disponibilité des programmes conçu pour une variété de systèmes d'exploitation, tels que Linux, FreeBSD, OpenBSD, Solaris et MacOS X. Distribué sous licence GPL, HeartBeat surveille les signaux de battements de cœur émis par les services au sein d'une grappe de serveurs pour évaluer leur opérationnalité.

En cas de défaillance détectée, il exécute des scripts d'initialisation, et une fois la disponibilité rétablie, il peut également effectuer des actions spécifiques. Ce logiciel offre la possibilité de changer d'adresse IP entre les machines en utilisant un mécanisme ARP, ce qui le rend adapté aux architectures réseau plus complexes. HeartBeat fonctionne efficacement avec deux machines, mais peut également être configuré pour des environnements réseau plus élaborés.



2. Préréquis

Nous aurons besoin des machines et configurations suivantes :

Machine 1 (srvweb1):

Host Name: srvWeb1

Carte réseau : LAN Segment sur VMWare

Adresse IP statique : 172.20.0.4/24

Machine 2 (srvweb2):

Hostname: srvWeb2

Carte réseau : LAN Segment sur VMWare

Adresse IP statique : 172.20.0.5/24

Machine 3 (Contrôleur de domaine):

Carte réseau : Segment LAN sur VMWare

Adresse IP statique : 172.20.0.1/24

Le contrôleur de domaine ne sert ici qu'à vérifier le bon fonctionnement du service Heartbeat

3. Réalisation

a. Installation des services apache & heartbeat

On va commencer par mettre à jour les distributions

```
root@srvWeb1:~# apt-get update && upgrade -y
```

```
root@srvWeb2:~# apt-get update && upgrade -y
```

Installer Apache2 et Heartbeat sur les machines Linux :

```
root@srvWeb2:~# apt install apache2|
```

```
root@srvWeb2:/var/www/html# apt install heartbeat|
```

Une fois le service apache installé, on va devoir configurer notre page web. En effet pour voir que le load balacing opère, nous allons devoir différencier nos deux pages webs en leur mettant un signe distinctif pour vérifier que notre service marche.

```
root@srvWeb2:~# rm /var/www/html/index.html
```

```
root@srvWeb2:~# cd /var/www/html
```

Télécharger un exemple de template bootstrap sur github :

```
root@srvWeb2:/var/www/html# wget https://github.com/technext/thegrill/archive/master.zip
```

On décompresse ensuite le fichier master.zip

```
root@srvWeb2:/var/www/html# unzip master.zip
```

Changez le document Root d'apache2 :

```
root@srvWeb2:/var/www/html# nano /etc/apache2/sites-available/000-default.conf|
```

Configurer les adresses ip des machines :

```
# The primary network interface
allow-hotplug ens33
iface ens33 inet static
address 172.20.0.5/24
```

```
# The primary network interface
allow-hotplug ens33
iface ens33 inet static
address 172.20.0.4/24
```

b. Configuration Heartbeat:

```
127.0.0.1    localhost
172.20.0.4  srvWeb1
172.20.0.5  srvWeb2
```

Suite à l'installation de heartbeat, un lien symbolique `/etc/heartbeat` est créé, pointant vers `/etc/ha.d`

Vous pouvez le vérifier à l'aide de la commande : `ls -l /etc/heartbeat`

Dans le dossier `/etc/ha.d`, nous allons créer 3 fichiers :

- **ha.cf** : fichier de configuration principal
- **haresources** : fichier de configuration des ressources.
- **authkeys** : information d'authentification

Vous devez créer les 3 fichiers avec le même contenu au niveau des 2 serveurs :

- **Premier fichier**

```
root@srvWeb1:~# nano /etc/ha.d/ha.cf|
```

Ajoutez les lignes suivantes :

Ajoutez les lignes suivantes :

```
# Les fichiers logs de heartbeat (événements relatifs à heartbeat)
```

```
logfile /var/log/heartbeat
```

```
logfacility local0
```

```
# Intervalle entre deux battements de cœur en seconde
```

```
keepalive 5
```

```
# Temps nécessaire avant de considérer qu'un serveur (noeud) est mort (en seconde)
```

```
deadtime 30
```

```
# Interface d'écoute
```

```
bcast ens33
```

```
# liste des nœuds utilisées pour la HD
```

```
node srvWeb1 srvWeb2
```

```
# comportement si le nœud revient dans le réseau
```

```
auto_failback on
```

- **Deuxième fichier :**

```
root@srvWeb2:~# nano /etc/ha.d/haresources|
```

Ajouter les lignes suivantes :

```
# Active l'interface IP Virtuelle avec comme noeud principal srvWeb1
```

```
# syntaxe :hostname IPaddr::IPvirtuelle/CIDR/interface service
```

```
srvWeb1 IPaddr::172.20.0.10/24/ens33 apache2
```

- **Troisième fichier :**

```
root@srvWeb1:~# nano /etc/ha.d/authkeys
```

Ajouter les lignes suivantes :

```
auth 1
```

```
1 md5 greta
```

Enregistrez le fichier, il faut sécuriser l'accès à ce dernier :

```
chmod 600 /etc/ha.d/authkeys
```

Désactivez au niveau des deux serveurs Web le démarrage automatique d'apache car c'est heartbeat (hairesources) qui va gérer le lancement de ce dernier, tapez la commande suivante :

```
root@srvWeb2:~# update-rc.d apache2 remove
```

Arrêtez apache2 :

```
root@srvWeb2:~# service apache2 stop
```

```
root@srvWeb1:~# service heartbeat stop
```

4. Vérification

Lancez Heartbeat et Apache2 sur srvWeb1, puis sur srvWeb2, en utilisant les commandes suivantes :

```
service apache2 start
```

```
service heartbeat start
```

Utilisez la commande ip ad pour vérifier la présence de l'interface virtuelle ens33 avec l'adresse 172.20.0.10 sur les machines srvWeb1 et srvWeb2.

À partir de la machine cliente, lancez Edge et accédez à l'adresse IP virtuelle : http://172.20.0.10. Vous devriez voir la page d'accueil d'Apache2, affichant "The Grill 1" !

Déconnectez l'interface réseau de srvWeb1 via les paramètres de la machine. Attendez un moment, puis utilisez la commande ip a sur srvWeb2. Vous devriez observer l'apparition de l'interface virtuelle 172.20.0.10, et le navigateur affichera maintenant "The Grill 2", indiquant que l'équilibrage de charge fonctionne correctement.

Si des erreurs surviennent, examinez les logs avec les commandes suivantes :