

Système de répartition de charge

Composants du système : 1 serveur Haproxy, 1 serveur NFS, et le nombre voulu de serveurs apache. Dans notre exemple nous en prendrons 2.

Dans un premier temps il faut installer et configurer haproxy :

Nous appellerons ce serveur **serveur A -> haproxy**
 serveur B -> Apache
 serveur C -> Apache
 serveur D -> NFS

A (haproxy) (en root) :

```
apt-get install haproxy
rm -f /etc/haproxy/haproxy.cfg
nano /etc/haproxy/haproxy.cfg
```

fichier haproxy.cfg :

```
global
    log 127.0.0.1 local0
    log 127.0.0.1 local1 notice
    #log loghost local0 info
    #connexion maximum par serveur à 4096
    maxconn 4096
    #debug
    #quiet
    #utilisateur
    user haproxy
    #groupe
    group haproxy

defaults
    log global
    # mode http ou https
    mode http
    #log http
    option httplog
    #permettra de loguer les connexions vides
    option dontlognull
    # restriction à 3 tentatives de connexions
    retries 3
    #autorise dans ce cas à renvoyer les connexions échouées vers le répartiteur
    redispatch
    #connexion maximum par serveur à 2000
    maxconn 2000
    #on abandonne si la connexion n'est pas établie après 5 secondes
    timeout 5000
    #temps d'attente d'une donnée de la part du client
    clienttimeout 50000
    #temps d'attente d'une donnée de la part du serveur
```

```

    srvtimeout    50000
#configuration du serveur haproxy (via son ip)
listen blweb 192.168.0.115:80
# mode http ou https (choix du service)
    mode http
# statistiques activée (panel)
    stats enable
#utilisateur du panel -> user / mot de passe -> password
    stats auth user:password
#répartition de charge entre les différents serveurs définis pour un service donné
    balance roundrobin
#Pour réutiliser un cookie applicatif et lui préfixer l'identifiant du serveur
    cookie JSESSIONID prefix
#cookie SERVERID (insert : cookie renseigné) (indirect : remplacer la valeur d'un cookie existant
#par celle attribuée à un serveur, lors d'un accès direct)
    cookie SERVERID insert indirect
#cookie SERVID prefix
#le serveur communiqueront sans "keep-alive HTTP"
    option httpclose
#permet à un serveur web final de connaître l'adresse IP du client initial
    option forwardfor
#message d'erreur du serveur si dé-commenter
    #option httpchk HEAD /haproxytest.txt HTTP/1.0
#serveur 1
    server web1 192.168.0.116:80 cookie A check
#serveur 2
    server web2 192.168.0.117:80 cookie B check

```

Puis enregistrer

```
/etc/init.d/haproxy restart
```

B (apache 1) (en root) :

```

apt-get install apache2 apache2-utils php5 php5-dev php5-gd -y
apt-get install mysql-server-5.5 php5-mysql -y
apt-get install apache2.2-common apache2-mpm-prefork apache2-utils ssl-cert -y
apt-get install libapache2-mod-php5 php5 php5-common php5-curl php5-gd php5-imagick php5-cli
-y
apt-get install mysql-server mysql-client libmysqlclient15-dev -y
apt-get install php5-mysql phpmyadmin -y

```

C (apache 2) (en root) :

```

apt-get install apache2 apache2-utils php5 php5-dev php5-gd -y
apt-get install mysql-server-5.5 php5-mysql -y
apt-get install apache2.2-common apache2-mpm-prefork apache2-utils ssl-cert -y
apt-get install libapache2-mod-php5 php5 php5-common php5-curl php5-gd php5-imagick php5-cli

```

```
-y
apt-get install mysql-server mysql-client libmysqlclient15-dev -y
apt-get install php5-mysql phpmyadmin -y
```

D (NFS) (en root) :

```
apt-get install apache2 apache2-utils php5 php5-dev php5-gd -y
apt-get install mysql-server-5.5 php5-mysql -y
apt-get install apache2.2-common apache2-mpm-prefork apache2-utils ssl-cert -y
apt-get install libapache2-mod-php5 php5 php5-common php5-curl php5-gd php5-imagick php5-cli
-y
apt-get install mysql-server mysql-client libmysqlclient15-dev -y
apt-get install php5-mysql phpmyadmin -y
apt-get install nfs-common nfs-kernel-server
nano /etc/exports
```

fichier /etc/exports (ajouter cette ligne en fin de fichier) :

```
/var/www IpserveurB(rw,sync,no_subtree_check)
/var/www IpserveurC(rw,sync,no_subtree_check)
```

Puis enregistrer.

```
exportfs -ra
/etc/init.d/nfs-kernel-server restart
```

Explications : Nous avons installer et configurer la machine faisant office de proxy (haproxy -> machine A)

Sur la machine B nous avons installer le serveur apache et ses dépendances.

Sur la machine C nous avons installer le serveur apache et ses dépendances.

Sur la machine D nous avons installer le serveur NFS (synchronisation de fichier d'autres machines) ce qui permet de ne pas mettre le contenu web sur chaque serveur apache.

Nous allons maintenant autoriser le serveur B et C (apaches) à aller sur le serveur NFS pour prendre les fichiers du contenu web de /var/www de la machine D (NFS). Puis configurer les apaches pour changer l'emplacement du contenu web.

B (apache 2) (en root) :

```
apt-get install nfs-common
mkdir /LieuDeMontageDuDossierNFS
```

nano /etc/fstab

fichier /etc/fstab (ajouter cette ligne en fin de fichier) :

IpserveurD:/var/www /LieuDeMontageDuDossierNFS nfs user,auto 0 0

Puis enregistrer

mount /LieuDeMontageDuDossierNFS

C (apache 2) (en root) :

apt-get install nfs-common

mkdir /LieuDeMontageDuDossierNFS

nano /etc/fstab

fichier /etc/fstab (ajouter cette ligne en fin de fichier) :

IpserveurD:/var/www /LieuDeMontageDuDossierNFS nfs user,auto 0 0

Puis enregistrer

mount /LieuDeMontageDuDossierNFS

Maintenant Haproxy disperse la charge du trafic web entre B et C et B et C vont chercher les fichiers sur D

Pour que B et C (les 2 serveurs apaches) prennent le contenu web de D (serveur nfs) il faut configurer les 2 apaches.

B & C (apache) (en root) :

Dans le fichier de configuration /etc/apache2/apache2.conf ajouter :

```
<Directory /LieuDeMontageDuDossierNFS>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

Puis enregistrer

fichier /etc/apache2/sites-available/000-default.conf

Modifier : DocumentRoot /var/www

Par : DocumentRoot /LieuDeMontageDuDossierNFS

Puis enregistrer.

ATTENTION SI VOUS AVEZ PLUSIEURS SITES SUR LE MÊME SERVEUR ALORS FAIRE COMME DANS LA MÉTHODE QUI SUIT :

fichier /etc/apache2/sites-avaible/000-default.conf

Mettre un '#' devant DocumentRoot :

```
#DocumentRoot /var/www
```

Puis ajouter à la fin du fichier (dans le cas de 2 sites) :

```
<VirtualHost *:80>
    DocumentRoot /LieuDeMontageDuDossierNFS/site1
    ServerName NomDeDomaineDuSite1
</VirtualHost>
```

```
<VirtualHost *:80>
    DocumentRoot /LieuDeMontageDuDossierNFS/site2
    ServerName NomDeDomaineDuSite2
</VirtualHost>
```

Puis enregistrer

```
/etc/init.d/apache2 restart
```

et de plus commenter la partie : <Directory /var/www/> ... </Directory>

Fin de la mise en place.