

# **COMPTE RENDU Haute disponibilité au sein d'un cluster.**

## Table des matières

Objectif(s) : .....	3
Installation du serveur Web principal : .....	3
a) Connexion au serveur Proxmox.....	3
b) Création de machine virtuelle : .....	3
c) Création de la deuxième machine virtuelle : .....	4
Mise en place de la haute disponibilité via Heartbeat: .....	4
a) Installation et configuration de Heartbeat sur le ServeurWeb1 :.....	5
b) Configuration de Heartbeat sur le Serveur Web2 :.....	7
TEST de la Haute disponibilité ! .....	7
Tableau récapitulatif du test de haute disponibilité selon plusieurs cas avec Heartbeat et les deux serveurs :.....	8

## Objectif(s) :

- Installer une machine virtuelle Debian8.2.
- Installer un serveur Web apache.
- Installer de la haute disponibilité logicielle via Heartbeat.
- Tester la continuité de service d'un serveur Web Apache.

## Installation du serveur Web principal :

a) Connexion au serveur Proxmox.

Tapez l'url suivant : <https://192.168.1.150:8006>

Id : root ;MDP : Salle105,

b) Création de machine virtuelle :

- *Voir Tuto création de VM sur PROXMOX VE.*
- *Une fois installé il faut penser à mettre à jour les paquets, avec un apt-get update.*
- Maintenant installer apache 2 et vérifier que le service s'exécute :

```
Apt-get install apache2<<
```

```
Service apache2 status
```

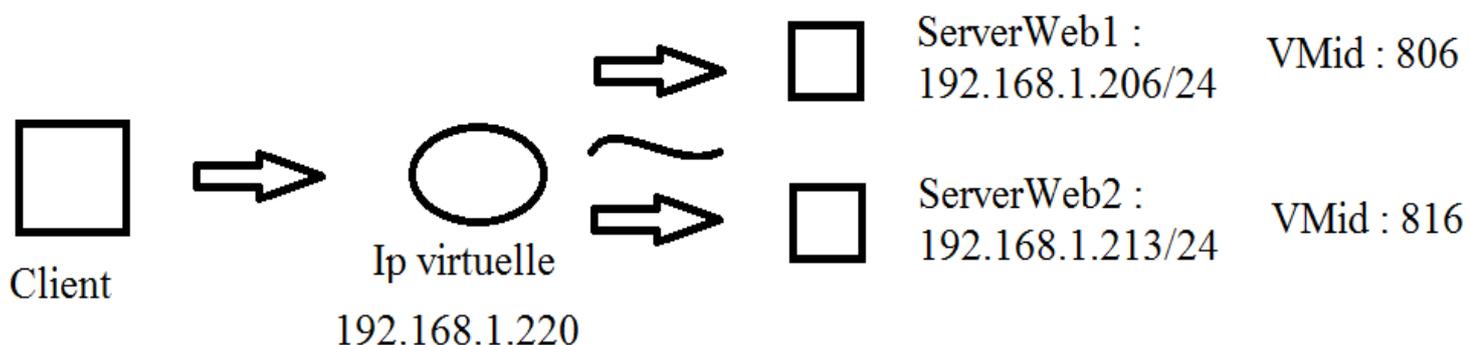
- *Modifier le fichier `index.html` dans `/var/www/html` et ajoutez la ligne suivante :*
  - `<h1>Serveur Web principal </h1>`
- *Testez à partir d'un navigateur Web :*
  - `127.0.0.1`

c) Création de la deuxième machine virtuelle :

- *Faire exactement pareil que la première machine,*
- *Modifier le fichier `index.html` dans `/var/www/html` et ajoutez la ligne suivante :*
  - `<h1>Serveur Web secondaire</h1>`

## Mise en place de la haute disponibilité via Heartbeat:

Schéma explicatif du projet « Haute disponibilité au sein d'un cluster Heartbeat :



~ Correspond au logiciel heartbeat

## a) Installation et configuration de Heartbeat sur le ServeurWeb1 :

- Installation de Heartbeat: apt-get install heartbeat.
- Configuration :
  - Dans /etc/heartbeat/ crée les 3 fichiers suivants :
    - Ha.cf
    - Haresources
    - Authkeys

## ➤ Dans le Ha.cf

- Ecrire cela :

```
logfile /var/log/ha-log
logfacility local0
keepalive 2
deadtime 10
bcast eth0
node serveurweb1 ServerWeb2L-H
auto_failback off
```

## ➤ Dans le haresources

- Ecrire cela :

```
serveurweb1 IPaddr::192.168.1.220/24/eth0
serveurweb1 apache2
```

## ➤ Dans le authkeys :

- Ecrire cela par exemple :

```
auth 3
3 md5 toinou
```

Pour avoir une sécurité et que personne ne puisse voir votre MDP, il suffit de restreindre les droits et de les mettre que sur celui qui a créé le document :

Chmod 600 /etc/heartbeat/authkeys

Pour que les noms d'hôtes présents dans les fichiers de configuration puissent être convertis en adresse IP, il faut configurer le fichier /etc/hosts.

➤ La configuration du /etc/hosts :

- Ecrire cela :

```
27.0.0.1    localhost
127.0.1.1   serveurweb1
192.168.1.206  serveurweb1
192.168.1.213  ServerWeb2L-H

# The following lines are desirable for IPv6 capable hosts
::1        localhost ip6-localhost ip6-loopback
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters
```

Quelques commandes :

- Service nom\_servicestart
- Service nom\_service stop
- Service nom\_servicestatus

Voir la liste des nœuds : cl\_statuslistnodes

Voir l'état d'un nœud : status nom-nœud

- On va maintenant éteindre apache 2 et faire en sorte que les services gérés par Heartbeat ne soient plus lancés automatiquement au démarrage de Linux :

Service apache 2 stop

Update-re.d -f nom\_service remove

*Il faut mettre les services gérés par Heartbeat et non le service  
Heartbeat !!!!*

b) Configuration de Heartbeat sur le Serveur Web2 :

Il faut faire exactement pareil,

Attention à mettre le bon nom de serveur aux emplacements demandé.

➤ On va pouvoir maintenant lancer Heartbeat sur les deux serveurs.

## **TEST de la Haute disponibilité !**

- Lorsqu'on accède d'un client Web à l'adresse IP virtuelle on est redirigé sur le serveur web !
- C'est la page Serveur Web principal index.html qui s'affiche.

Tableau récapitulatif du test de haute disponibilité selon plusieurs cas avec Heartbeat et les deux serveurs :

	ServeurWeb1 défaillant	ServeurWeb1 opérationnel	ServeurWeb2 défaillant	ServeurWeb2 opérationnel
Que se passe-t-il ?	On bascule sur le ServeurWeb2 grâce à Heartbeat.	On retourne sur le ServeurWeb1 grâce à Heartbeat (auto_failback on)	On reste sur le ServeurWeb1 comme c'est lui le « maître ».	On reste sur le ServeurWeb1 comme c'est lui le « maître ».
La page Web index.html est-elle disponible ?	La page Web index.html est disponible car on se situe sur le ServeurWeb2 grâce au basculement fait par Heartbeat.	<b>X</b>	La page Web index.html est disponible car on se situe sur le ServeurWeb1 qui est le « maître ».	<b>X</b>
Commandes adéquates pour expliquer ce qui se passe	cl_status hbstatus cl_status listnodes cl_status <u>nodestatus</u> <u>nom_noeud</u>	cl_status hbstatus cl_status listnodes cl_status <u>nodestatus</u> <u>nom_noeud</u>	cl_status hbstatus cl_status listnodes cl_status <u>nodestatus</u> <u>nom_noeud</u>	cl_status hbstatus cl_status listnodes cl_status <u>nodestatus</u> <u>nom_noeud</u>

La mise en œuvre de la haute disponibilité via Heartbeat permet de faire une continuité de services.

Le problème qui subsiste si le ServeurWeb1 crash malgré la mise en place d'une solution de haute disponibilité est qu'il n'y a pas de synchronisation des données entre les serveurs.

Pour résoudre ce problème, il faut mettre en place une solution de synchronisation des données comme avec la commandersync.