

## Sommaire :

Introduction.....	1
I - Installation du serveur Web principal : .....	2
II - Installation du serveur Web secondaire : .....	2
III – Mise en place de la haute disponibilité via Heartbeat : .....	2
IV – Test de la haute disponibilité : .....	3
V – Conclusion : .....	5

## Introduction

C'est un TP.

Groupe	Noeud	Adresse IP	VMID	Adresse IP Debian 8.2	Nom de la VM
Anthony	PVE3	192.168.1.152	803	192.168.1.203	ServeurWeb1DUPONT
Sébastien	PVE3	192.168.1.152	813	192.168.1.210	ServeurWeb2DETROZ
				<b>IP VIRTUELLE :</b> 192.168.1.217	

L'installation sur le NAS est plus rapide que l'installation en local.

E6 :

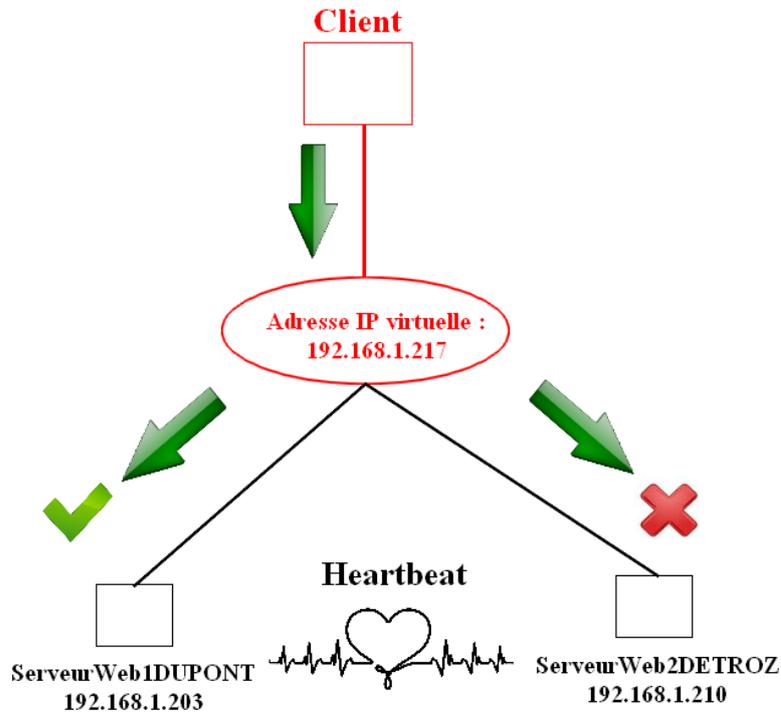
Elaboration de documents relatifs à la production et à la fourniture de services

A1.1.1 , Analyse du cahier des charges d'un service à produire

A1.2.4 , Détermination des tests nécessaires à la validation d'un service

A1.3.4 , Déploiement d'un service

A4.1.9 , Rédaction d'une documentation technique



## I - Installation du serveur Web principal :

Pour l'installation de la machine, il suffit de suivre le TP, tout est précisé.

## II - Installation du serveur Web secondaire :

Pour l'installation de la machine, il suffit de suivre le TP, tout est précisé.

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

Il y a 3 fichiers nécessaires à la configuration de Heartbeat :

- **ha.cf** qui est le fichier de configuration principale. Ce fichier détermine la liste des machines à utiliser et la manière de dialoguer entre elles.
- **haresources** qui est le fichier de configuration de ressources. Ce fichier indique les opérations à effectuer au démarrage de la haute disponibilité sur une machine.
- **authkeys** qui est le fichier d'information de l'authentification. Ce fichier détermine la clé et le protocole de protection utilisé.

**ha.cf :**

```
logfile      /etc/heartbeat/log/ha-log
logfacility   local0
keepalive    2
deadtime     10
bcast       eth0
node        ServeurWeb1DUPONT ServeurWeb2DETROZ
auto_failback off
```

**haresources :**

ServeurWeb1DUPONT IPaddr::192.168.1.217/24/eth0 apache2

**authkeys :**

```
auth 3
3 md5 chalazion
```

On protège le fichier authkeys avec : **chmod 600 /etc/heartbeat/authkeys**

On arrête le service Web actuel avec : **service apache2 stop**

On désactive le lancement automatique d'Apache au démarrage : **update-rc.d -f apache2 remove**

On remplit ensuite le fichier **/etc/hosts** :

```
192.168.1.203 ServeurWeb1DUPONT
192.168.1.210 ServeurWeb2DETROZ
```

Après avoir démarré Heartbeat sur les deux serveurs, les interfaces

## IV – Test de la haute disponibilité :

Situation	ifconfig	cl_status hbstatus	cl_status listnodes	cl_status nodestatus nom_du_serveur	ping	Page affichée
-----------	----------	-----------------------	------------------------	---	------	---------------

---

Haute disponibilité de service via Heartbeat

---

<b>Le ServeurWeb2 est défaillant</b>	Le ServeurWeb1 a l'adresse virtuelle	Heartbeat est actif sur le ServeurWeb1	Les deux serveurs sont toujours présents	Le ServeurWeb1 est actif / Le ServeurWeb2 est dead	La continuité de service est assurée	Page du serveur principal
<b>Reprise d'activité du ServeurWeb2</b>	Le ServeurWeb1 a l'adresse virtuelle	Heartbeat est actif sur les deux serveurs	Les deux serveurs sont toujours présents	Les deux serveurs sont actifs	La continuité de service est assurée	Page du serveur principal
<b>Le ServeurWeb1 est défaillant</b> (auto_failback : OFF)	Le ServeurWeb2 a l'adresse virtuelle	Heartbeat est actif sur le ServeurWeb2	Les deux serveurs sont toujours présents	Le ServeurWeb1 est dead / Le ServeurWeb2 est actif	Perte d'un ping	Page du serveur secondaire
<b>Reprise d'activité du ServeurWeb1</b> (auto_failback : OFF)	Le ServeurWeb2 a l'adresse virtuelle	Heartbeat est actif sur les deux serveurs	Les deux serveurs sont toujours présents	Les deux serveurs sont actifs	La continuité de service est assurée	Page du serveur secondaire
<b>Le ServeurWeb1 est défaillant</b> (auto_failback : ON)	Le ServeurWeb2 a l'adresse virtuelle	Heartbeat est actif sur le ServeurWeb2	Les deux serveurs sont toujours présents	Le ServeurWeb1 est dead / Le ServeurWeb2 est actif	Perte d'un ping	Page du serveur secondaire
<b>Reprise d'activité du ServeurWeb1</b> (auto_failback : ON)	Le ServeurWeb1 a l'adresse virtuelle	Heartbeat est actif sur les deux serveurs	Les deux serveurs sont toujours présents	Les deux serveurs sont actifs	Perte d'un ping	Page du serveur principale

## V – Conclusion :

La haute disponibilité permet d'assurer la disponibilité d'un service en redondant celui-ci. Le service est donc toujours opérationnel malgré la coupure d'un des serveurs qui le prodiguait.

Les données qui sont sauvegardées sur un serveur ne seront pas sauvegardées sur l'autre si celui-ci est éteint. Le problème de la synchronisation des données est donc toujours présent.

Il faut donc implanter un service de synchronisation des données.

---

Le service est assuré de manière permanente et les changements de serveurs sont invisibles pour l'utilisateur. (La continuité de service est dégradée si le serveur de secours est moins puissant que le serveur principal).

Tous les fichiers sauvegardés sur le serveur principal ne sont pas disponibles sur le serveur secondaire.

Il faut donc mettre en place un système de synchronisation automatique des données d'un serveur vers un autre. Des solutions possibles sont rsync sous Linux, ou DRBD, pour pouvoir mettre à jour les données.