

Documentation : Mise en place d'un HAProxy sur Cluster pfSense (Load Balancing de GLPI)

Sommaire

1. Préparation des serveurs GLPI (Clonage Proxmox)
 2. Configuration réseau et application des serveurs
 3. Installation du package HAProxy sur pfSense (Master et Slave)
 4. Configuration du Backend (Les Ouvriers)
 5. Configuration du Frontend (La Porte d'Entrée)
 6. Tests et Vérifications
-

1. Préparation des serveurs GLPI (Proxmox)

Objectif : Créer deux serveurs web identiques hébergeant ton application GLPI, prêts à travailler en parallèle.

- **Pourquoi ?** Pour garantir que les deux serveurs fournissent exactement la même application, la même configuration système, et les mêmes fichiers. C'est la base de la haute disponibilité.
1. **Sur Proxmox :** Fais un clic droit sur ta VM GLPI originale (SRV_glpi1).
 2. **Choisis Clone.**
 3. **Mode : Full Clone.**
 - **Pourquoi ?** Un clone complet crée une copie autonome de tous les disques virtuels de la VM. Il est totalement indépendant du parent.
 4. **Nomme-la : SRV_glpi2.**
 5. **Valide** et laisse Proxmox terminer.
 6. **Ressources :** Vérifie que SRV_glpi2 a les mêmes ressources (CPU, RAM) que SRV_glpi1 pour assurer des performances homogènes.
-

2. Configuration réseau et application des serveurs

Objectif : Donner une identité unique à chaque serveur sur le réseau et synchroniser l'accès à la base de données.

- **Chaque serveur doit être sur le même réseau (ex : vmbr1, le réseau serveurs).**

Sur SRV_glpi2 (Le Clone) :

1. **Démarre la VM SRV_glpi2.**
2. **Modifie l'IP :**
 - **Comment ?** Édite `/etc/netplan/*.yaml` (Ubuntu) ou `/etc/network/interfaces` (Debian).
 - **Nouvelle IP :** Mettez 192.168.10.5 (en supposant que SRV_glpi1 est en .10.3).
 - **Masque :** /24 (255.255.255.0).
3. **Modifie le nom d'hôte :**
 - Édite `/etc/hostname` pour mettre glpi2.
 - Édite `/etc/hosts` pour mettre à jour la ligne associée.
4. **Synchronisation critique :** Redémarre la VM pour appliquer les changements : reboot.

Sur SRV_glpi1 ET SRV_glpi2 (Les Deux Serveurs) :

1. **Configuration Base de Données :**
 - Édite le fichier de configuration de GLPI :
`/var/www/html/glpi/config/config_db.php`.
 - **Important :** Dans un environnement HA, les deux serveurs doivent pointer vers la **même base de données centralisée** (ou un cluster de base de données). Si tu les fais pointer vers 127.0.0.1 (local), tes données ne seront pas synchronisées (les tickets créés sur le serveur 1 n'apparaîtront pas sur le serveur 2).
 - **Correction :** Change l'hôte de la base de données pour l'IP de ton **Cluster de Base de Données** (ex : 192.168.10.10) pour qu'ils utilisent une base de données centralisée et synchronisée.
-

3. Installation du package HAProxy sur pfSense (Master et Slave)

Objectif : Installer le service de proxy inverse sur les deux membres de ton cluster pour garantir la haute disponibilité.

- **Pourquoi installer sur les deux ?** Puisque tu utilises CARP pour la haute disponibilité de pfSense, le service HAProxy doit être disponible et configuré de

manière identique sur le Master et le Slave pour que le service web continue sans interruption si le Master tombe.

1. Connecte-toi sur pfSense MASTER (via son IP d'administration, pas la VIP).

- Va dans System > Package Manager > Available Packages.
- Cherche haproxy.
- Clique sur Install et confirme.

2. Connecte-toi sur pfSense SLAVE (via son IP d'administration).

- **Répète exactement la même procédure :** Va dans System > Package Manager > Available Packages.
- Cherche haproxy, installe et confirme.

4. Configuration du Backend (Le groupe de serveurs)

Objectif : Définir le pool de serveurs web (SRV_glpi1, SRV_glpi2) qui traiteront les requêtes.

- 1. Connecte-toi sur pfSense MASTER.**
2. Va dans Services > HAProxy > Backend et clique sur Add.
3. Name : GLPI_Backend.
4. Load Balancing Algorithm : Choisis Round Robin.
 - **Pourquoi ?** C'est un algorithme simple et équilibré qui distribue les requêtes de manière équitable sur chaque serveur du pool.

Server List (Le tableau) :

Nom	Adresse	Port	Cookie
-----	---------	------	--------

SRV1	192.168.10.3	80	SRV1
------	--------------	----	------

SRV2	192.168.10.5	80	SRV2
------	--------------	----	------

Exporter vers Sheets

- Note : Pour voir la case "Cookie", clique sur l'ancre à gauche de la ligne.
5. Health check method : Choisis HTTP.
 - **Pourquoi ?** C'est une vérification de santé plus précise qu'un simple test de port TCP. pfSense effectuera une requête HTTP GET vers / (ou une page

spécifique, ex : /index.php). Si le serveur répond avec un code 200 OK, pfSense le considère comme opérationnel.

6. Cookie persistence :

- Coche Cookie Enabled.
- Cookie Name : SERVERID.
- Cookie Mode : Insert.
- **Pourquoi ?** Pour que l'utilisateur reste sur le même serveur durant toute sa session. C'est indispensable pour GLPI, qui utilise des sessions PHP locales. Si l'utilisateur change de serveur à chaque requête, sa session sera perdue. Le mode Insert crée et envoie un cookie au client pour mémoriser le serveur initial.

7. Clique sur Save.

5. Configuration du Frontend (L'IP publique/virtuelle)

Objectif : Définir l'adresse IP virtuelle sur laquelle HAProxy écoute les requêtes des clients.

1. **Toujours sur pfSense MASTER.**

2. Va dans Services > HAProxy > Frontend et clique sur Add.

3. Name : GLPI_Frontend.

4. External Address :

- Listen Address : Choisis ta **VIP CARP d'administration** (ex : 192.168.20.254). C'est l'IP que les clients taperont.
- Port : 80. (C'est le port HTTP standard).

5. Type : http / https(offloading).

- **Pourquoi ?** Cette option permet à pfSense de gérer le trafic HTTP standard et, si configuré ultérieurement, de gérer le trafic HTTPS offloading (certificat SSL installé sur pfSense, trafic en HTTP entre pfSense et les serveurs). Pour un BTS, c'est une configuration robuste.

6. Default Backend : Sélectionne ton GLPI_Backend.

Advanced Settings :

1. Coche Use "forwardfor" option.

- **Pourquoi ?** HAProxy agit comme un reverse proxy. Sans cette option, tes serveurs GLPI verraient l'IP de pfSense comme source de toutes les requêtes. forwardfor ajoute l'en-tête HTTP X-Forwarded-For contenant la véritable IP de l'utilisateur. C'est indispensable pour les logs de sécurité et d'audit de GLPI.
- *Pense à configurer Apache (remoteip_module) ou Nginx (real_ip module) sur les serveurs GLPI pour lire cet en-tête.*

ACLs spécifiques (Crucial pour GLPI API) :

Si tu utilises l'API GLPI (ex : pour FusionInventory/GLPI-Agent), ces agents gèrent mal les cookies de session. Il faut désactiver la persistance pour ces chemins.

1. Dans la section ACL, clique sur Add.
2. **Ligne 1** : Name: GLPI_API_Path | Expression: path_beg | CS: non-sensible | Value: /plugins/fusioninventory/ /api/.
 - **Pourquoi ?** CS (non-sensible) pour ne pas différencier /API de /api.
 - **Value** : Coche les deux chemins possibles.
3. En dessous, dans la section Actions : clique sur Add.
4. **Ligne 1** : Action: Use Backend | parameters: GLPI_Backend | Condition acl: GLPI_API_Path.
5. Clique sur Save, puis sur le bouton vert Apply Changes.

6. Tests et Vérifications

Objectif : Valider le bon fonctionnement du Load Balancing et de la persistance.

Vérifier le statut :

- Sur pfSense MASTER, va dans Stats (ou Services > HAProxy > Stats).
- Frontend : Doit être VERT (Open).
- Backend : Les deux serveurs doivent être VERTS (UP).

Le test de charge :

1. **Depuis un PC client**, tape la VIP (ex : 192.168.20.254 ou le nom DNS associé).
2. GLPI doit s'afficher. Connecte-toi.
3. **Vérification de la persistance :**

- Utilise un navigateur en mode "Privé" ou un autre PC. Connecte-toi à GLPI via la VIP.
- Dans les outils de développement (F12) > Onglet Réseau (Network), vérifie les en-têtes de réponse. Tu dois voir le cookie SERVERID. Note sa valeur (SRV1 ou SRV2).
- Ouvre une nouvelle fenêtre en mode "Privé". Tu devrais voir le cookie pointant vers l'autre serveur. C'est la preuve que la persistance par cookie fonctionne.

Test de panne (Serveurs) :

1. **Sur Proxmox**, éteins proprement la VM SRV_glpi1.
2. **Sur pfSense MASTER (page Stats)**, SRV1 doit passer en rouge (DOWN) après quelques secondes. Le backend GLPI_Backend doit rester VERT (UP).
3. **Actualise la page sur le client** : GLPI doit toujours fonctionner, car HAProxy a basculé automatiquement sur le serveur 2 (glpi2).
4. **Redémarre SRV_glpi1**. Il doit repasser en VERT (UP) sur pfSense.

Test de panne (pfSense Master) :

1. **Éteins proprement pfSense MASTER.**
2. **Sur pfSense SLAVE**, la VIP CARP doit basculer sur lui (System > High Availability Sync). Le service HAProxy doit prendre le relais.
3. **Actualise la page sur le client** : GLPI doit toujours fonctionner via la VIP.