

Tuto : Mise en place d'une machine de cache pour Steam, Riot Games, Blizzard

Documentation du projet :

<https://github.com/steamcache/generic>

<https://github.com/steamcache/steamcache-dns>

Prérequis :

- Un routeur sous Debian de la WAN
- Machine de cache, sous linux (Ubuntu 18.4 LTS serveur) avec un disque (un SSD est mieux) pouvant contenir tous les jeux (install complète. Compter ~15Go pour CSGO, ~7Go pour Rocket League, ~8Go pour LoL, ~30Go pour Overwatch, ~3Go pour Hearthstone) et de la RAM suffisante (8Go) une carte Gigabit Ethernet pour supporter la Bande passante

Installation :

Sur le router

Choisir des adresses statiques pour le serveur, il faut compter une adresse pour le cache Steam, une pour le cache Blizzard, une pour le cache Riot Games et une pour l'adresse générique du cache. Ceci fait donc un total de 4 adresses. Comme nous avons un portail captif, il faut libérer ces adresses choisies. Pour ce faire taper la commande :

```
wanparty-add-machine <@IP> <nom> ORGA
```

Il faut donc la taper 4 fois, avec nos 4 adresses choisies et un nom approprié.

Dans mon cas

```
wanparty-add-machine 10.26.0.10 steam-cache-dns ORGA
```

```
wanparty-add-machine 10.26.0.11 cache-steam ORGA
```

```
wanparty-add-machine 10.26.0.12 cache-blizzard ORGA
```

```
wanparty-add-machine 10.26.0.13 cache-riotGames ORGA
```

Ensuite pour que le cache fonctionne, il faut changer les bords DHCP. En effet la machine fonctionne sur des requêtes DNS pour mettre en cache ce qui vient des serveurs de Steam, Blizzard, Riot Games (ou d'autres suivant comment on la configure).

Pour cela dans le fichier `/etc/dhcp/dhcpd.conf`, on modifie la ligne concernant les DNS :

```
option domain-name-servers 10.26.0.10, 10.26.0.1 ;
```

Pour la prise en compte des nouveaux paramètres restarter le service

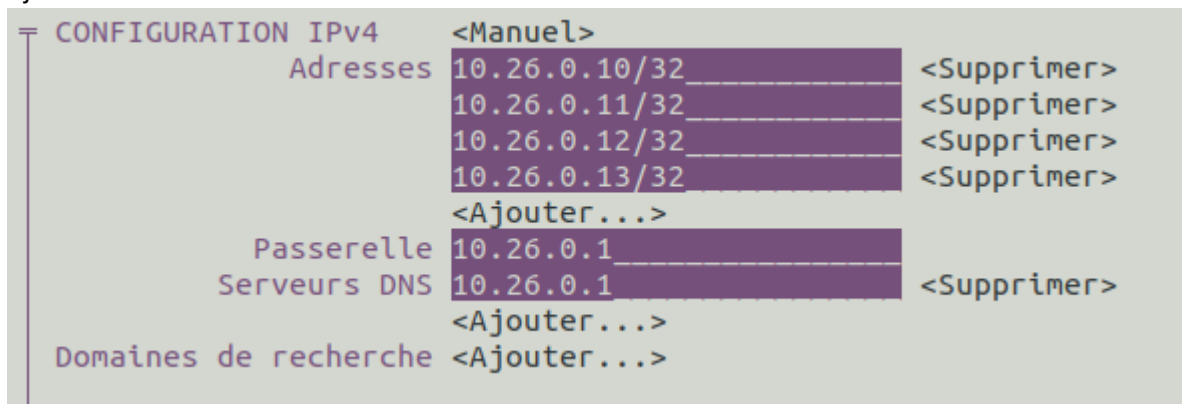
Pour la machine de cache

On affecte des paramètres ip statique sur la machine avec la commande :

```
sudo nmtui
```

Puis a l'ouverture des menus, on va dans :

- Modifier une connexion
- Sélection de l'interface
- Ajout des adresses IP et du DNS :



- Puis on valide
- Puis retour
- Et quitter

Ensuite on restart le service networking (`/etc/init.d/networking restart`)

On vérifie que les adresses ont bien été ajoutées (`hostname -I` par exemple). Si les adresses non pas été changer faire un reboot de la machine.

Puis on met à jour la liste des packages :

```
Sudo apt-get update
```

Puis on install curl pour pouvoir installer docker

```
sudo apt install curl -y
```

```
sudo curl -ssl https://get.docker.com/ | sh
```

Puis on regarde que docker est bien installer

```
sudo systemctl status docker
```

Maintenant on peut ajouter des containers. Pour ce faire, il faut en faire un pour chaque éditeur de jeux.

```
sudo docker run \  
--restart unless-stopped -d \  
--name cache-steam \  
-v /home/steam-cache/cache/steam/data:/data/cache \  
-v /home/steam-cache/cache/steam/logs:/data/logs \  
-p 10.26.0.11:80:80 \  
steamcache/generic:latest
```

Pour détail :

--name indique le nom de votre container

-v permet de renseigner un chemin pour les logs (fin avec:/data/logs) et un chemin pour les données qui vont être télécharger à la suite des requêtes DNS suite a la demande du téléchargent d'un jeu ou d'une mise à jour.

-p permet d'indiquer l'adresse IP que l'on va utiliser pour ce container. Le :80 :80 est important

Donc on effectue la même chose pour les autres.

Pour Blizzard

```
sudo docker run \  
--restart unless-stopped -d \  
--name cache-blizzard \  
-v /home/steam-cache/cache/blizzard/data:/data/cache \  
-v /home/steam-cache/cache/blizzard/logs:/data/logs \  
-p 10.26.0.12:80:80 \  
steamcache/generic:latest
```

Pour Riot Games

```
sudo docker run \  
--restart unless-stopped -d \  
--name cache-riotGames \  
-v /home/steam-cache/cache/riotGames/data:/data/cache \  
-v /home/steam-cache/cache/riotGames/logs:/data/logs \  
-p 10.26.0.13:80:80 \  
steamcache/generic:latest
```

Pour finir, on fait le dernier container, celui qui permet de regrouper en une seule adresse les trois autres. Pour ce faire, on tape:

```
sudo docker run \  
--restart unless-stopped -d \  
--name steam-cache-dns \  
-p 10.26.0.10:53:53/udp \  
-e USE_GENERIC_CACHE=true \  
-e LANCACHE_IP=10.26.0.10 \  
-e UPSTREAM_DNS=10.26.0.1 \  
-e STEAMCACHE_IP=10.26.0.11 \  
-e BLIZZARDCACHE_IP=10.26.0.12 \  
-e RIOTCACHE_IP=10.26.0.13 \  
-e CACHE_MEM_SIZE=5000m \  
-e CACHE_DISK_SIZE=100g \  
steamcache/steamcache-dns:latest
```

De même ici, le `--name` permet de donner un nom au container

`-p` une adresse IP, important de bien mettre `:53 :53/udp`

`-e` permet de renseigner les adresses IP des autres containers, l'utilisation du cache générique et de définir la taille du disque utiliser ainsi que de la RAM

Voilà tout est prés maintenant,

Pour vérifier que les containers sont bien démarrés, on utilise la commande :

```
sudo docker ps -a
```

Pour voir les logs, et surtout s'assurer que les donner sont bien télécharger depuis notre machine, on peut taper la commande

```
tail -f /home/cache-steam/cache/steam/logs/access.log
```

Pour le cache de Steam dans cet exemple, il suffit de changer le chemin pour les autres.

Vous remarquerez que nous n'avons pas renseigner de fichier de logs pour le cache générique, il n'y a donc pas de fichier de logs ? Si si, il y en a bien un, il faut taper la commande :

```
sudo docker logs -f steam-cache-dns
```