

## CONTEXTE DU PROJET

The Grill est une chaîne de restaurants haut de gamme spécialisée dans les grillades et la cuisine raffinée, disposant de plusieurs établissements répartis sur le territoire national. Fort de son succès et de sa notoriété croissante, The Grill a développé une **plateforme web complète** permettant à ses clients de consulter les menus, découvrir les restaurants, réserver une table en ligne et commander des plats à emporter ou en livraison.

Avec l'augmentation constante du trafic web (notamment lors des heures de pointe de réservation entre 18h et 20h, ainsi que lors d'événements spéciaux et promotions), l'infrastructure web actuelle, basée sur un **serveur unique**, montre ses limites. L'entreprise fait face à des **ralentissements significatifs** aux heures de forte affluence, et le serveur unique constitue un **point de défaillance unique (SPOF - Single Point Of Failure)** critique : toute panne matérielle, surcharge ou maintenance planifiée entraîne l'indisponibilité totale du site web, impactant directement les réservations et donc le chiffre d'affaires.

Pour répondre à ces défis et accompagner la croissance de l'entreprise, la direction de The Grill a décidé de moderniser son infrastructure informatique en mettant en place une solution d'**équilibrage de charge (load balancing)** basée sur **HAProxy**. Cette solution open source, reconnue pour sa fiabilité et ses performances exceptionnelles, est largement utilisée par les sites à fort trafic dans le monde. HAProxy permettra de répartir intelligemment les requêtes HTTP entre plusieurs serveurs web, d'assurer la haute disponibilité en cas de panne d'un serveur, et de fournir des statistiques en temps réel pour la supervision de l'infrastructure.

### CONTEXTE DU PROJET

The Grill est une chaîne de restaurants haut de gamme spécialisée dans les grillades et la cuisine raffinée, disposant de plusieurs établissements répartis sur le territoire national. Fort de son succès et de sa notoriété croissante, The Grill a développé une plateforme web complète permettant à ses clients de consulter les menus, découvrir les restaurants, réserver une table en ligne et commander des plats à emporter ou en livraison.

Avec l'augmentation constante du trafic web (notamment lors des heures de pointe de réservation entre 18h et 20h, ainsi que lors d'événements spéciaux et promotions), l'infrastructure web actuelle, basée sur un serveur unique, montre ses limites. L'entreprise fait face à des ralentissements significatifs aux heures de forte affluence, et le serveur unique constitue un point de défaillance unique (SPOF - Single Point Of Failure) critique : toute panne matérielle, surcharge ou maintenance planifiée entraîne l'indisponibilité totale du site web, impactant directement les réservations et donc le chiffre d'affaires.

Pour répondre à ces défis et accompagner la croissance de l'entreprise, la direction de The Grill a décidé de moderniser son infrastructure informatique en mettant en place une solution d'**équilibrage de charge (load balancing)** basée sur **HAProxy**. Cette solution open source, reconnue pour sa fiabilité et ses performances exceptionnelles, est largement utilisée par les sites à fort trafic dans le monde. HAProxy permettra de répartir intelligemment les requêtes HTTP entre plusieurs serveurs web, d'assurer la haute disponibilité en cas de panne d'un serveur, et de fournir des statistiques en temps réel pour la supervision de l'infrastructure.

### Introduction :

HAProxy est une solution gratuite, très rapide et fiable offrant la haute disponibilité, équilibrage de charge et proxy pour les applications TCP et HTTP. Il convient particulièrement aux sites Web à très fort trafic et alimente bon nombre des sites les plus visités au monde. Au fil des années, il est devenu l'équilibreur de charge opensource standard. Il est désormais livré avec la plupart des distributions Linux classiques et est souvent déployé par défaut sur des platesformes cloud.

### Remarque :

Attention : il y'a deux segments LAN sur VMWare "LAN" et "WAN"

Une VM HAProxy de répartition de charge : hostname :

HAProxy (apt install haproxy )

network adapter : carte 1 : en segment LAN "LAN": static 172.20.0.20/24 network adapter 1 :  
carte 2 : en segment LAN "WAN": static 192.168.0.20/24

Deux serveurs Web ( apt install apache2 php wget unzip ) :

srvWeb1 :

- nom d'hôte : SRV WEB1
- adaptateur réseau : carte 1 172.20.0.21/24 : en segment LAN "LAN" : statique

srvWeb2 :

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
iface enp0s3 inet static
    address 172.20.0.20
    netmask 255.255.255.0

auto enp0s8
iface enp0s8 inet static
    address 192.168.0.20
    netmask 255.255.255.0
```

· nom d'hôte : SRV WEB2

- adaptateur réseau : carte 1 172.20.0.22/24 : en segment LAN "LAN" : statique

Un client machine Windows) :

Carte réseau1 : en segment LAN "WAN" : statique : 192.168.0.100/24

Personnalisez la page d'accueil d'apache2 :

Personnaliser les pages d'accueil d'apache2, au niveau des 2 serveurs Web rm  
/var/www/html/index.html

Accédez au documentRoot d'apache cd

/var/www/html

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Oct 21 15:34:34 CEST 2024 on tty1
invite@debian:~$ su
Mot de passe :
root@debian:/home/invite# apt install haproxy_
```

Téléchargez l'exemple de Template suivant :

```
wget https://github.com/technext/thegrill/archive/master.zip
```

Décompressez le fichier zip :

```
unzip master.zip
```

Changez le document Root d'apache2 : nano  
/etc/apache2/sites-available/000-default.conf

Enregistrez puis redémarrez Apache2 service

```
apache2 restart
```

Dans le fichier /var/www/html/thegrill-master/index.html, ajouter respectivement 1 et 2 après le titre THE GRILL des serveurs Web 1 et

Validation 1 :

Au niveau de toutes les machines, vérifiez la configuration IP :

"ip a" et "ipconfig"

Depuis la machine HAProxy :

```
· ping 172.20.0.21      ok
· ping 172.20.0.22      ok
·
·
```

Depuis les deux serveurs Web : service apache2 status

En cours d'exécution

Depuis la machine cliente : ping

```
192.168.0.20    OK
```

Configuration de HAProxy :

Nous allons configurer le service haproxy pour assurer le balancement (load balancing) entre les serveurs web du cluster.

Le fichier de configuration de haproxy est /etc/haproxy/haproxy.cfg, éditez ce dernier (nano) pour ajouter à la fin du fichier les lignes suivantes :

```
nano /etc/haproxy/haproxy.cfg
```

Ajouter le lignes suivantes à la fin du fichier :

## Configuration du balancement

```
listen clusterWebStadiumCompany bind
```

```
192.168.0.20 80
```

## **mode d'écoute**

mode http

## **mode du balancement (roundrobin 50% 50%)**

balance roundrobin

## **Option**

option httpclose option  
forwardfor

## **Liste des serveurs impliqués par le balancement**

server SRV WEB1 172.20.0.21 80 check server SRV  
WEB2 172.20.0.22 80 check

## **Pour les statistiques**

stats enable stats hide-version  
stats refresh 30s stats show-  
node stats auth  
admin:password stats uri  
/statistique Enregistrez votre  
fichier.

Validation :

Pour vérifier le balancement : RDV machine client IE ou Edge 192.168.0.20, vous devez voir The Grill 1, actualisez, vous devez voir The Grill 2, le balancement est bien fonctionnel.

Pour vérifier les stats : RDV machine client IE 192.168.0.20/statistique, le login c'est admin:password, vous devez voir une page de statistique avec les deux serveurs Web up en vert.

Pour tester, vous pouvez "service apache2 stop" sur l'un des serveurs, attendre 30 secondes et vérifiez que le serveur en question passe en état down en rouge dans la page des statistiques.

Explications :

listen cluster\_web adresseIPHAproxy:80 Cette directive nous permet de spécifier sur quelle adresse IP HaProxy va fonctionner, nous allons accéder au contenu web depuis cette adresse IP.

mode http: permet de spécifier que le balancement de charge est utilisé pour du contenu web http, dans le cas contraire on peut utiliser le mode tcp (pour du mysql par exemple ;)

balance roundrobin: permet de spécifier l'algorithme de répartition de charge. Il en existe plusieurs

**RoundRobin:** La méthode Round-robin est une répartition équitable de la charge entre les serveurs d'un cluster. Chaque serveur traite le même nombre de requêtes, mais cela nécessite d'avoir des serveurs homogènes en termes de capacité de traitement.

**Source :** Le mode de balancement « source » signifie qu'un client en fonction de son adresse IP sera toujours dirigé vers le même serveur web. Cette option est nécessaire lorsque les sites Internet utilisent des sessions PHP.

**Least connection:** Le serveur renvoie vers le serveur le moins chargé. Si en théorie il semble le plus adapté, en réalité dans le cadre du Web dynamique, un serveur peut être considéré comme chargé alors que les processus sont en attente d'une requête vers une base de données.

**First Response:** Les requêtes clients sont envoyées simultanément à tous les serveurs et le premier qui répond sera chargé de la connexion. Difficile à mettre en oeuvre et rarement employé. "

Bien sûr ! Voici un compte-rendu complet, rédigé comme si tu venais de réaliser la configuration complète de HAProxy pour la haute disponibilité et le load balancing, avec une conclusion à la fin. Tu pourras facilement l'intégrer dans un document Word/LibreOffice, ajouter tes captures d'écran, puis exporter en PDF.

## **Compte-rendu : Mise en place de HAProxy pour la haute disponibilité et le load balancing**

### **1. Introduction**

HAProxy est une solution open source très performante et fiable, utilisée pour assurer la haute disponibilité et l'équilibrage de charge (load balancing) des applications TCP et HTTP. Elle est particulièrement adaptée aux sites à fort trafic et est largement déployée dans le monde professionnel.

Dans ce TP, j'ai configuré HAProxy sur une VM Debian pour répartir la charge entre deux serveurs web Apache.

### **2. Architecture et contexte**

- **VM HAProxy**
  - Hostname : HAProxy
  - 2 interfaces réseau :
    - LAN (segment LAN : statique 172.20.0.20/24)
    - WAN (segment WAN : statique 192.168.0.20/24)
- **Serveurs Web**
  - SRV-WEB1 : 172.20.0.21/24 LAN
  - SRV-WEB2 : 172.20.0.22/24 LAN
- **Client Windows)**
  - Carte réseau en segment WAN 192.168.0.100/24

*(Insérer un schéma réseau illustrant les connexions entre client, HAProxy, et les deux serveurs web)*

### **3. Préparation des serveurs web**

#### **3.1 Installation Apache2 et PHP**

Sur chaque serveur web (SRV-WEB1 et SRV-WEB2), j'ai installé Apache2 et PHP :

#### **3.2 Personnalisation des pages d'accueil**

Pour différencier visuellement les deux serveurs, j'ai personnalisé la page d'accueil :

J'ai modifié le fichier `/var/www/html/thegrill-master/index.html` pour ajouter un numéro distinctif 1 pour SRV WEB1, 2 pour SRV WEB2) dans le titre **THE GRILL**.

#### **3.3 Configuration du DocumentRoot**

J'ai modifié le fichier Apache par défaut pour que le DocumentRoot pointe vers le dossier `thegrill-master` :

J'ai remplacé la ligne :

par :

Puis j'ai redémarré Apache :

### **4. Validation réseau**

### **5. Installation et configuration de HAProxy**

#### **5.1 Installation**

Sur la VM HAProxy, j'ai installé HAProxy :

#### **5.2 Configuration**

J'ai édité le fichier `/etc/haproxy/haproxy.cfg` pour ajouter la configuration suivante à la fin :

Ajout :

#### **5.3 Redémarrage de HA Proxy**

## 6. Validation du load balancing

### 6.1 Test de répartition

Depuis la machine cliente, j'ai ouvert un navigateur et accédé à :

À chaque actualisation, la page affichait alternativement **THE GRILL 1** puis **THE GRILL 2**, confirmant que le load balancing en mode round-robin fonctionnait correctement.

### 6.2 Accès aux statistiques

J'ai accédé à :

avec le login `admin` et le mot de passe `password`.

La page de statistiques affichait les deux serveurs en vert (up), avec leur état et nombre de connexions.

### 6.3 Test de tolérance aux pannes

J'ai arrêté Apache sur un des serveurs :

Après environ 30 secondes, la page de statistiques indiquait que ce serveur était en rouge (down), et HAProxy redirigeait automatiquement tout le trafic vers l'autre serveur disponible.

## 7. Explications techniques

`listen clusterWebStadiumCompany` : définit une section d'écoute sur l'IP

- `192.168.0.20 port 80. mode http` : HAProxy traite les requêtes HTTP.
- `balance roundrobin` : répartit les requêtes de manière équitable entre les serveurs.
- `option httpclose` : ferme la connexion HTTP après chaque requête.
- `option forwardfor` : transmet l'adresse IP du client aux serveurs backend.
- `server ... check` : définit les serveurs backend avec vérification de leur disponibilité.
- **Statistiques** : interface web protégée pour surveiller l'état des serveurs.
- 
- 
- **8. Conclusion**

Ce TP m'a permis de mettre en place un équilibrEUR de charge fiable avec HAProxy, assurant la haute disponibilité de deux serveurs web Apache. Le load balancing round-robin fonctionne parfaitement, répartissant les requêtes entre les serveurs. La surveillance via l'interface de statistiques permet de détecter rapidement les pannes et d'assurer une continuité de service.

HAProxy s'est révélé être un outil puissant, simple à configurer, et essentiel pour gérer des architectures web à fort trafic.