

HAPROXY

Emetteur(s) : Matthieu SAVIARD

Destinataire(s) : CFA INSTA

Date : 03/12/2023

Objet : Mise en production d'un serveur HAPROXY

1. Présentation

HAProxy est une solution open source, réputée pour sa robustesse, sa rapidité et sa fiabilité. Il sert de pilier essentiel dans la gestion de la haute disponibilité, l'équilibrage de charge et le proxy pour les applications TCP et HTTP. Cette plateforme trouve particulièrement son utilité dans le contexte de sites web à fort trafic, étant devenue le choix standard en matière d'équilibreur de charge open source.

Principales fonctionnalités :

- Équilibrage de charge avancé : HAProxy optimise les performances en répartissant la charge entre plusieurs serveurs, offrant ainsi une expérience utilisateur fluide même en cas de trafic intense. Il propose divers algorithmes d'équilibrage, tels que Round Robin, Least Connections, etc.
- Proxy TCP/HTTP : En agissant en tant que proxy pour les protocoles TCP et HTTP, HAProxy permet une gestion souple du trafic. Il peut également assurer la terminaison SSL pour les connexions sécurisées HTTPS.
- Haute disponibilité (HA) : HAProxy garantit la disponibilité continue des applications en redirigeant le trafic vers des serveurs opérationnels. Il effectue des vérifications de santé régulières pour s'assurer que les serveurs sont opérationnels.
- Configuration flexible : La configuration de HAProxy, réalisée à travers des fichiers texte, offre une flexibilité optimale. Cela permet aux administrateurs de définir des règles complexes en fonction des exigences spécifiques de leur infrastructure.

- Performance élevée : Renommé pour ses performances exceptionnelles et sa faible latence, HAProxy peut gérer un grand nombre de connexions simultanées tout en maintenant une efficacité opérationnelle élevée.
- Open Source : Distribué sous licence open source (GPL), HAProxy est une solution gratuite, libre à la modification et à la distribution. Intégration cloud : HAProxy s'intègre aisément avec des environnements cloud tels que AWS, Azure et Google Cloud, ce qui en fait un choix privilégié pour les déploiements dans le cloud.

Documentation exhaustive : HAProxy dispose d'une documentation complète, facilitant ainsi la compréhension et la configuration pour les utilisateurs, qu'ils soient novices ou experts. En somme, HAProxy demeure un choix incontournable pour les architectures distribuées, les environnements cloud et les sites web à fort trafic en raison de sa fiabilité, de ses performances éprouvées et de sa flexibilité d'intégration.

2. Prérequis

3 machines Debian 12 :

- 1 machine HAPROXY
- 2 serveurs webs

Machine	OS	IP
SERVWEB-03	Debian 12	172.20.0.6
SERVWEB-04	Debian 12	172.20.0.7
HAPROXY	Debian 12	172.20.0.8

3. Réalisation

a. SERVWEB-03

On commence par renommer la machine

```
root@DNS-01:~# hostnamectl set-hostname servweb3 _
```

On reboot ensuite la machine

On va ensuite mettre à jour la machine

```
root@servweb3:~# apt update
```

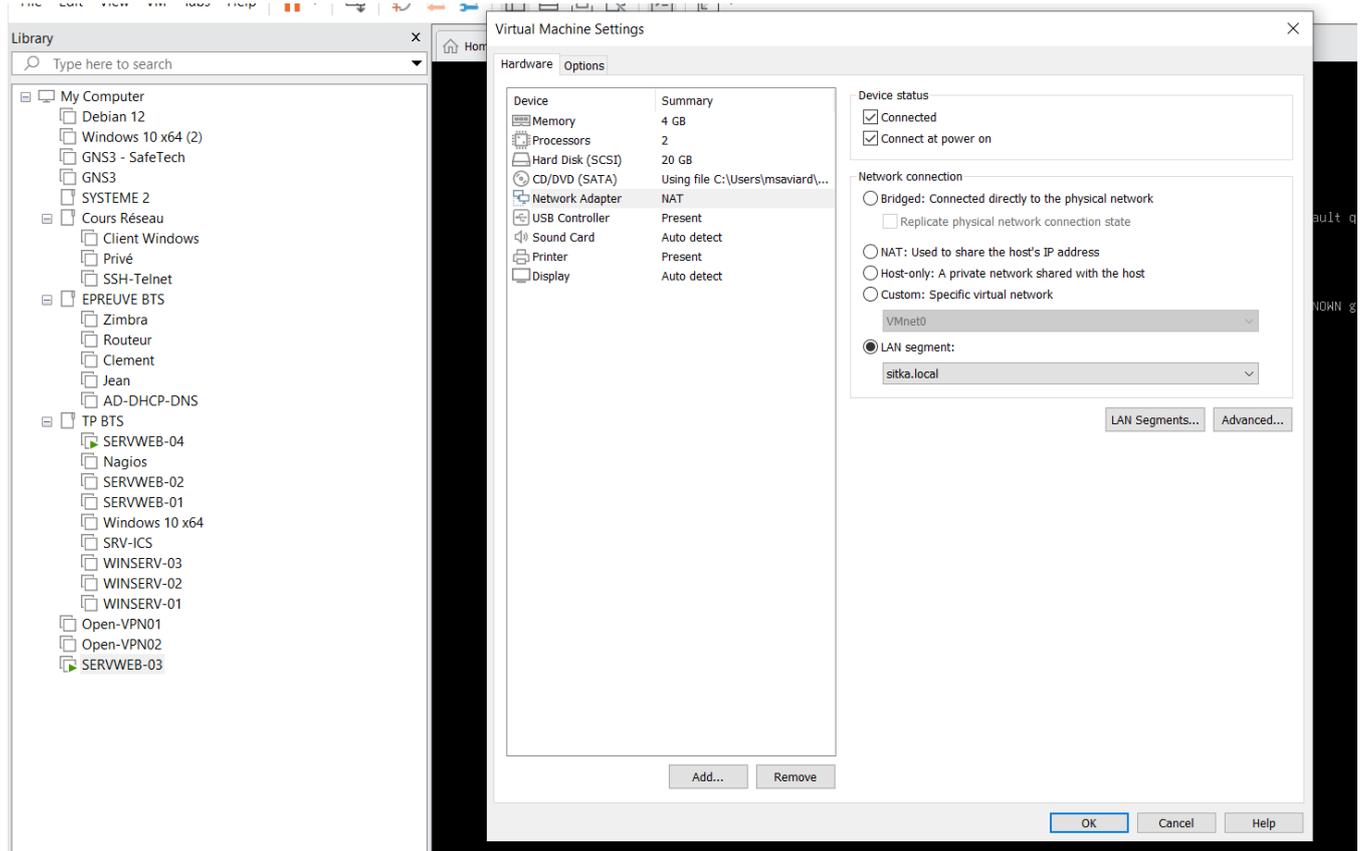
```
root@servweb3:~# apt upgrade
```

On va ensuite configurer les interfaces et les mettre en statique avec l'IP 172.20.0.6

```
root@servweb3:~# nano /etc/network/interfaces_
```

```
GNU nano 7.2 /etc/network/  
# This file describes the network interfaces available  
# and how to activate them. For more information, see  
  
source /etc/network/interfaces.d/*  
  
# The loopback network interface  
auto lo  
iface lo inet loopback  
  
# The primary network interface  
allow-hotplug ens33  
iface ens33 inet static  
address 172.20.0.6/24
```

On va ensuite dans les configurations VMware et on met la machine en LAN SEGMENT avec la LAN sitka.local



On redémarre ensuite les interfaces de la machine

```
valid_lft forever preferred_lft forever
root@servweb3:~# service networking restart
root@servweb3:~# ip ad
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:96:a1:9f brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.6/24 brd 172.20.0.255 scope global ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe96:a19f/64 scope link tentative
        valid_lft forever preferred_lft forever
root@servweb3:~# _
```

On voit donc que l'IP a bien changé. Nous allons donc pouvoir installer apache2

```
root@servweb3:~# apt install apache2 -y _
```

On va ensuite modifier le site et télécharger un site depuis github

```
root@servweb3:~# cd /var/www/*
```

On va donc une fois dans le dossier, télécharger le fichier

```
root@servweb3:/var/www/html# wget https://github.com/technext/thegrill/archive/master.zip
```

On va télécharger la commande unzip pour unzipper le fichier zip téléchargé

```
root@servweb3:/var/www/html# apt install unzip
```

On va unzip le fichier pour avoir le dossier

```
root@servweb3:/var/www/html# unzip master.zip
```

On supprime la page par défaut de apache

```
root@servweb3:/var/www/html# rm index.html
```

Et ensuite on va dans le fichier de configuration de notre serveur apache pour le faire pointer vers notre page web située dans /var/www/html/thegrill-master/index.html

```
root@servweb3:/var/www/html# nano /etc/apache2/sites-available/000-default.conf
```

Une fois sur le fichier, rajouter the grill master derrière /var/www/html

```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html/thegrill-master
```

On va ensuite reboot le service apache

```
root@servweb3:/var/www/html# service apache2 restart
```

On va aller ensuite sur le fichier html de the grill master et rajouter un 3 pour identifier la machine quand le load balancing opère :

```
root@servweb3:/var/www/html# cd thegrill-master/
root@servweb3:/var/www/html/thegrill-master# ls
404.html assets index.html index-video.html single-post.html
root@servweb3:/var/www/html/thegrill-master# nano index.html
```

```
root@servweb3:/var/www/html/thegrill-master# nano index.html
```

On remplace le <title> par <title> Serveur Web 03 </title>

```
user@servweb3: ~
GNU nano 7.2 index.html *
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="">
  <meta name="author" content="">
  <link rel="shortcut icon" href="assets/img/ico/favicon.ico">
  <link rel="apple-touch-icon" sizes="144x144" href="assets/img/ico/apple-touch-icon-144x144.png">
  <link rel="apple-touch-icon" sizes="114x114" href="assets/img/ico/apple-touch-icon-114x114.png">
  <link rel="apple-touch-icon" sizes="72x72" href="assets/img/ico/apple-touch-icon-72x72.png">
  <link rel="apple-touch-icon" href="assets/img/ico/apple-touch-icon-57x57.png">

  <title>Serveur Web 03</title>

  <!-- Bootstrap Core CSS -->
  <link href="assets/css/bootstrap.min.css" rel="stylesheet">
  <link href="assets/css/animate.css" rel="stylesheet">
  <link href="assets/css/plugins.css" rel="stylesheet">

  <!-- Custom CSS -->
  <link href="assets/css/style.css" rel="stylesheet">

  <!-- Custom Fonts -->
  <link href="assets/font-awesome-4.1.0/css/font-awesome.min.css" rel="stylesheet" type="text/css">
  <link href="assets/css/pe-icons.css" rel="stylesheet">
```

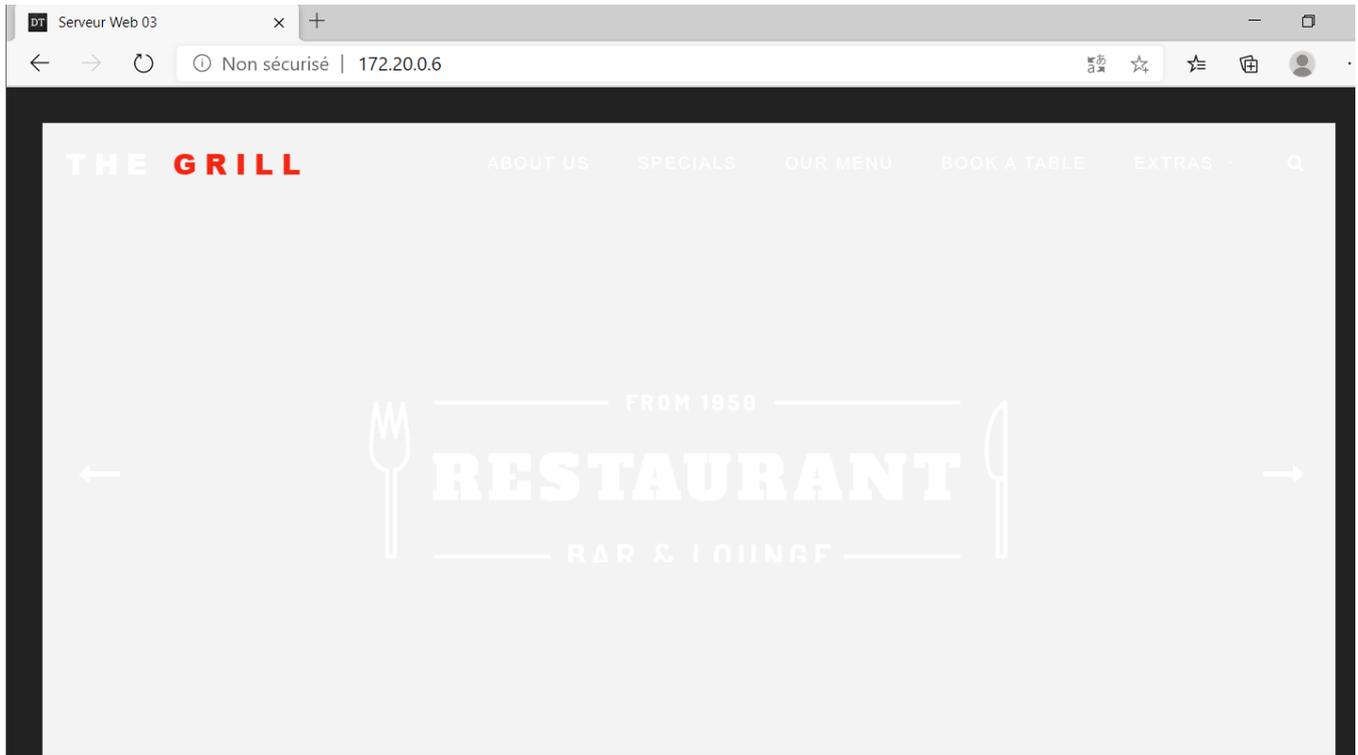
```
  <a class="navbar-brand smoothie" href="index.html">THE <span class="theme-accen
</div>

<!-- Collect the nav links, forms, and other content for toggling -->
<div class="collapse navbar-collapse" id="main-navigation">
  <ul class="nav navbar-nav navbar-right">
    <li><a href="#about" class="page-scroll">Serveur Web 03</a></li>
    <li><a href="#about" class="page-scroll">Specials</a></li>
    <li><a href="#the-menu" class="page-scroll">Dur Menu</a></li>
    <li><a href="#about" class="page-scroll">Book a Table</a></li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-hover="dropdown" data-toggle="
      <ul class="dropdown-menu" role="menu">
        <li><a href="404.html">404</a></li>
      </ul>
    </li>
    <li><a href="#search"><i class="fa fa-search"></i></a></li>
  </ul>
</div>
```

On restart le service

```
root@servweb3:/var/www/html# service apache2 restart
```

On va ensuite sur un contrôleur de domaine sur le même VLAN et taper <http://172.20.0.6> sur internet voir si la page web est bien visible



La page s'affiche, le serveur est donc bien opérationnel, On va donc pouvoir configurer le prochain serveur web.

b. SERVWEB-04

On commence par renommer la machine

```
root@DNS-01:~# hostnamectl set-hostname servweb4  
root@DNS-01:~#
```

On reboot ensuite la machine

On va ensuite mettre à jour la machine

```
root@servweb3:~# apt update
```

```
root@servweb3:~# apt upgrade
```

On va ensuite configurer les interfaces et les mettre en statique avec l'IP 172.20.0.7

```
root@servweb3:~# nano /etc/network/interfaces_
```

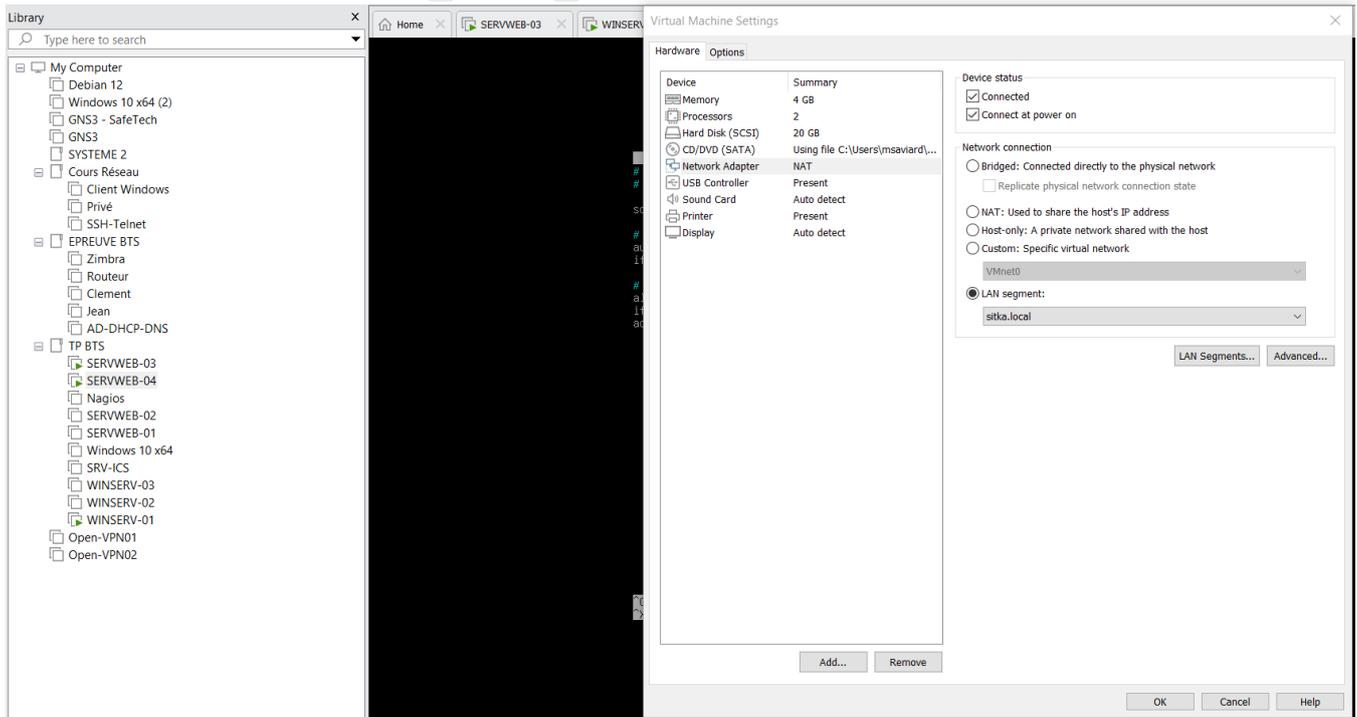
```
GNU nano 7.2 /etc/network/interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug ens33
iface ens33 inet static
address 172.20.0.7/24
```

On va ensuite dans la configuration VMware et on change la VLAN pour sitka.local



On restart ensuite l'interface

```
root@servweb4:~# service networking restart _
```

On voit donc que l'IP a bien changée. Nous allons donc pouvoir installer apache2

```
root@servweb3:~# apt install apache2 -y _
```

On va ensuite modifier le site et télécharger un site depuis github

```
root@servweb3:~# cd /var/www/*
```

On va donc une fois dans le dossier, télécharger le fichier

```
root@servweb3:/var/www/html# wget https://github.com/technext/thegrill/archive/master.zip
```

On va télécharger la commande unzip pour unzipper le fichier zip téléchargé

```
root@servweb3:/var/www/html# apt install unzip
```

On va unzip le fichier pour avoir le dossier

```
root@servweb3:/var/www/html# unzip master.zip
```

On supprime la page par défaut de apache

```
root@servweb3:/var/www/html# rm index.html
```

Et ensuite on va dans le fichier de configuration de notre serveur apache pour le faire pointer vers notre page web située dans /var/www/html/thegrill-master/index.html

```
root@servweb3:/var/www/html# nano /etc/apache2/sites-available/000-default.conf
```

Une fois sur le fichier, rajouter the grill master derrière /var/www/html

```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html/thegrill-master
```

On va ensuite reboot le service apache

```
root@servweb3:/var/www/html# service apache2 restart
```

On va aller ensuite sur le fichier html de the grill master et rajouter un 3 pour identifier la machine quand le load balancing opère :

```
root@servweb3:/var/www/html# cd thegrill-master/
root@servweb3:/var/www/html/thegrill-master# ls
404.html assets index.html index-video.html single-post.html
root@servweb3:/var/www/html/thegrill-master# nano index.html
```

```
root@servweb3:/var/www/html/thegrill-master# nano index.html
```

On remplace le <title> par <title> Serveur Web 04 </title>

```
<!DOCTYPE html>
<html lang="en">

<head>

  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="">
  <meta name="author" content="">
  <link rel="shortcut icon" href="assets/img/ico/favicon.ico">
  <link rel="apple-touch-icon" sizes="144x144" href="assets/img/ico/apple-touch-icon-144x144.png">
  <link rel="apple-touch-icon" sizes="114x114" href="assets/img/ico/apple-touch-icon-114x114.png">
  <link rel="apple-touch-icon" sizes="72x72" href="assets/img/ico/apple-touch-icon-72x72.png">
  <link rel="apple-touch-icon" href="assets/img/ico/apple-touch-icon-57x57.png">

  <title>Serveur Web 04</title>

  <!-- Bootstrap Core CSS -->
  <link href="assets/css/bootstrap.min.css" rel="stylesheet">
  <link href="assets/css/animate.css" rel="stylesheet">
  <link href="assets/css/olueins.css" rel="stylesheet">
```

```

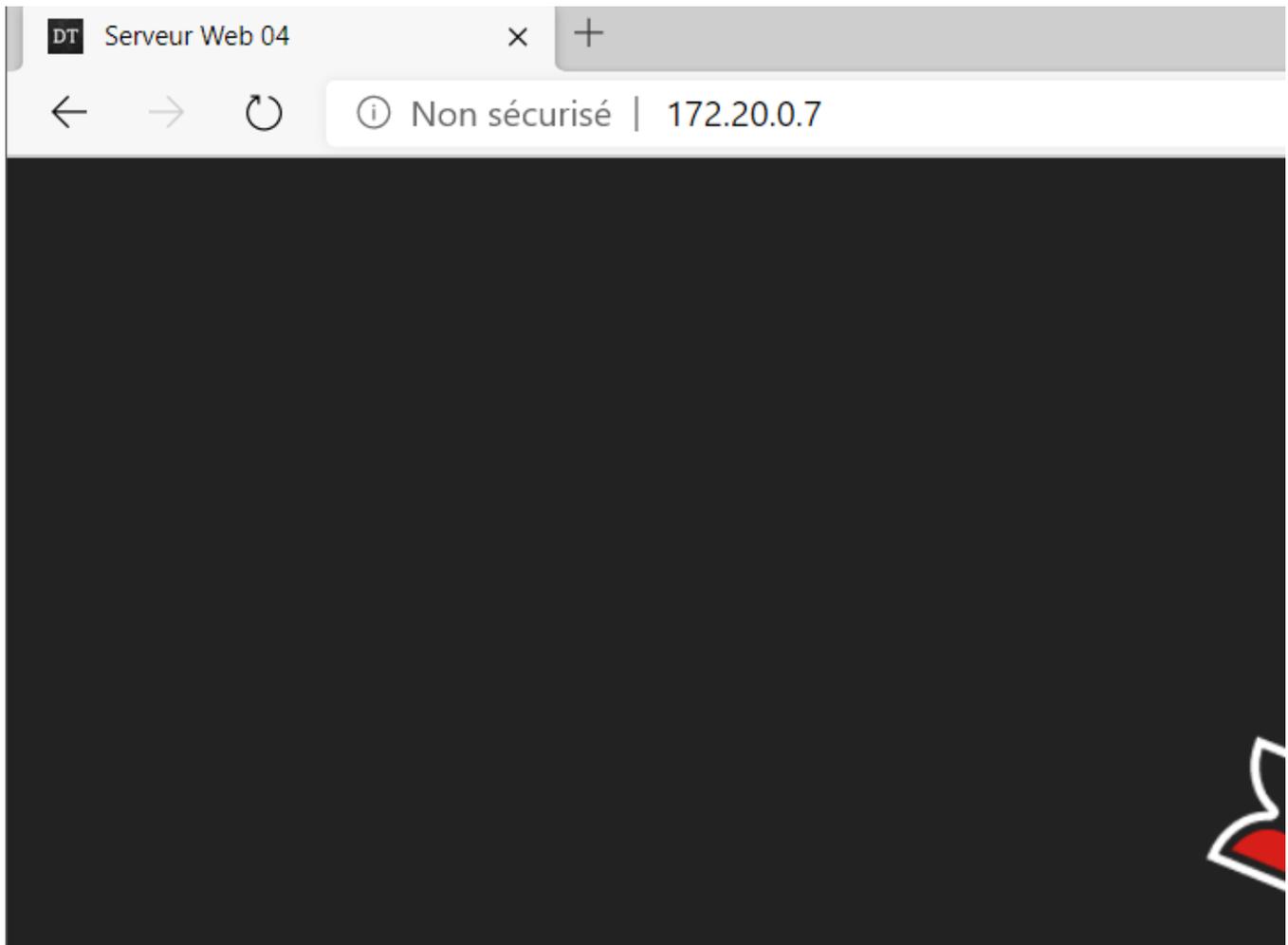
<!-- Collect the nav links, forms, and other content for toggling -->
<div class="collapse navbar-collapse" id="main-navigation">
  <ul class="nav navbar-nav navbar-right">
    <li><a href="#about" class="page-scroll">SERVEUR 04</a></li>
    <li><a href="#about" class="page-scroll">Specials</a></li>
    <li><a href="#the-menu" class="page-scroll">Our Menu</a></li>
    <li><a href="#about" class="page-scroll">Book a Table</a></li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-hover="dropdown" data-toggle="dropdown">
        <ul class="dropdown-menu" role="menu">
          <li><a href="404.html">404</a></li>
        </ul>
      </a>
    </li>
    <li><a href="#search"><i class="fa fa-search"></i></a></li>
  </ul>
</div>

```

On restart le service

```
root@servweb3:/var/www/html# service apache2 restart
```

On va ensuite sur un contrôleur de domaine sur le même VLAN et taper <http://172.20.0.7> sur internet voir si la page web est bien visible



Le site est bien visible, nous allons donc pouvoir configurer HAProxy

c. HAPROXY

INTERFACE	IP	TYPE
ENS33	172.20.0.7	LAN
ENS36	DHCP	NAT

On va donc configurer le HAPROXY de la façon suivante :

Device	Summary
Memory	4 GB
Processors	2
Hard Disk (SCSI)	20 GB
CD/DVD (SATA)	Using file C:\Users\msaviard\...
Network Adapter	NAT
Network Adapter 2	LAN Segment
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Une en NAT et une en LAN

On commence par renommer la machine

```
root@DNS-01:~# hostnamectl set-hostname haproxy
```

Puis reboot

On met cette configuration réseau

```
GNU nano 7.2 /etc/ne
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see the man page of
# the /etc/network/interfaces file.

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug ens33
iface ens33 inet dhcp

# The secondary network interface
auto ens37
iface ens37 inet static
address 172.20.0.8/24
```

Puis on met à jour les paquets.

```
root@haproxy:~# apt update && upgrade _
```

On va ensuite commencer à installer HAPROXY :

```
root@haproxy:~# apt install haproxy
```

Une fois le paquet installé, on va éditer le fichier de configuration `/etc/haproxy/haproxy.cfg`

```
root@haproxy:~# nano /etc/haproxy/haproxy.cfg
```

Et rajouter les lignes suivantes à la fin

```

# Configuration du balancement
listen clusterWebSitka
bind 192.168.32.143:80
# 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 srvweb3 172.20.0.6:80 check
server srvweb4 172.20.0.7:80 check
# Pour les statistiques
stats enable
stats hide-version
stats refresh 30s
stats show-node
stats auth admin:password
stats uri /statistique

```

On redémarre ensuite le service

```

root@haproxy:~# service haproxy restart
root@haproxy:~# service haproxy status
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
   Active: active (running) since Sun 2023-12-03 14:58:31 CET; 5s ago
     Docs: man:haproxy(1)
           file:/usr/share/doc/haproxy/configuration.txt.gz
   Main PID: 870 (haproxy)
     Tasks: 3 (limit: 4915)
    Memory: 41.2M
         CPU: 101ms
   CGroup: /system.slice/haproxy.service
           └─870 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
             872 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

déc. 03 14:58:31 haproxy systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
déc. 03 14:58:31 haproxy haproxy[870]: [NOTICE] (870) : New worker (872) forked
déc. 03 14:58:31 haproxy haproxy[870]: [NOTICE] (870) : Loading success.
déc. 03 14:58:31 haproxy systemd[1]: Started haproxy.service - HAProxy Load Balancer.
root@haproxy:~#

```

On peut voir qu'il tourne bien. La configuration est donc faite et opérationnelle

4. Vérification

Pour évaluer le fonctionnement de l'équilibrage, veuillez procéder comme suit : depuis la machine cliente, utilisez le navigateur Internet Explorer ou Edge pour accéder à l'adresse IP 192.168.32.143. Vous devriez observer l'affichage de "The Grill 3". Après actualisation, vous devriez alors constater l'affichage de "The Grill 4". La confirmation de cette alternance démontre le bon fonctionnement de l'équilibrage.

