TP 5 - HAProxy

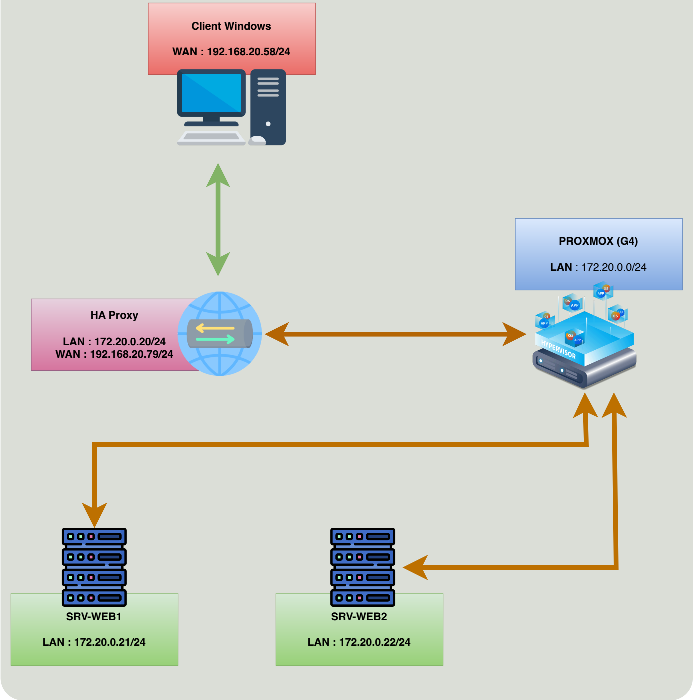
Contexte du TP

L'entreprise **AlphaTech Services**, spécialisée dans la gestion d'infrastructures informatiques pour des clients PME, souhaite améliorer la **disponibilité** et la **continuité de service** de son site web d'entreprise, accessible depuis Internet.

Pour se faire, l'équipe d'exploitation choisit de mettre en place une architecture de répartition de charge (**load balancing**) et de **haute disponibilité** en utilisant la solution open source **HAProxy** reconnue pour sa robustesse.

L'intérêt de ce TP est de déployer et configurer cette solution dans un environnement virtualisé afin de simuler les conditions réelles d'un réseau d'entreprise.

Topologie du réseau



Qu'est-ce qu'un Reverse Proxy ?

Un Reverse Proxy, ou bien serveur mandataire inverse, est un type de serveur qui est habituellement placé en frontal de serveurs web.
Contrairement au serveur proxy qui permet à un utilisateur d'accéder au réseau Internet, le reverse proxy est utilisé par les serveurs pour gérer et filtrer les requêtes entrantes.

En sommes, il reçoit toutes les requêtes et décide vers quel serveur backend les envoyer, et renvoie ensuite la réponse au client.

Les fonctionnalités :
- **Sécurisation** : masquage des adresses réelles des serveurs, filtre du trafic,

termine les connexions HTTPS ;
- **Performance** : mise en cache du contenu, compression des réponses, persistance de session si nécessaire ;
- **Scalabilité** : fait du load balancing entre plusieurs serveurs et permet un déploiement derrière une seule adresse IP publique.

Pour résumer, un reverse proxy joue le rôle de porte d'entrée pour une application web en combinant sécurité, performance et distribution du trafic.

Que permet la solution HAProxy ?

HAProxy (High Availability Proxy) est un logiciel open-source très performant utilisé comme **reverse proxy** et surtout comme **load balancer** pour des applications web, des API ou tout type de service TCP/HTTP.

Ces fonctionnalités principales sont :
- **Répartition de charge** (Load Balancing) : distribue les requêtes entre plusieurs serveurs backends, plusieurs algorithmes disponibles (round robin, leastconn, source hashing, etc) ;
- **Haute disponibilité** : détection automatique des serveurs en panne, retire et réintègre les serveurs dynamiquement, peut fonctionner en mode actif/passif ;
- **Reverse Proxy HTTP et TCP** : gère tous les types de trafic et contrôle fin des règles de routage et du trafic ;
- **Accélération des performances** : cache de réponses (avec certaines configurations) ;
- **La sécurité** : filtrage, contrôle d'accès, rate limiting, protection contre certaines attaques (DDoS) ;

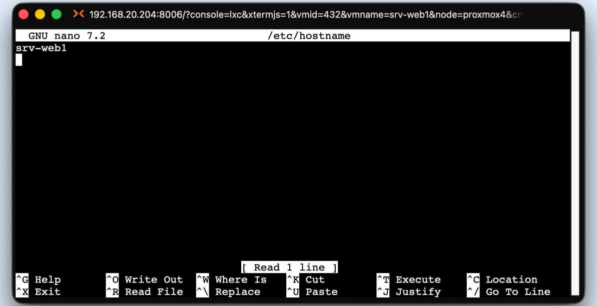
Étape 0 : Création et préparation des machines

Réalisez ces étapes pour vos deux serveurs.

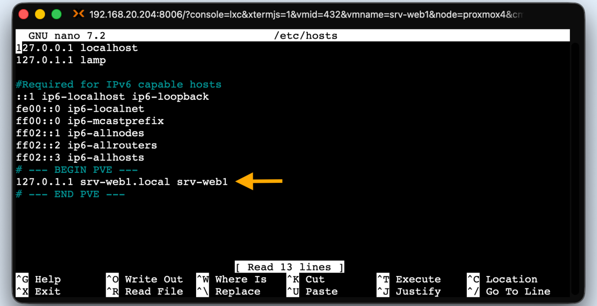
1.1 - Changement du hostname

Créez deux machines virtuelles sous Debian et changez leur hostname puis redémarrez les machines.

SRV-WEB1

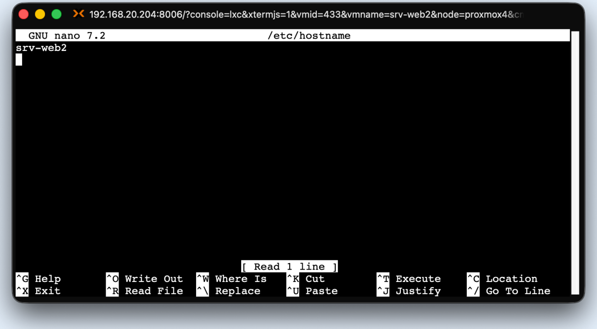

192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=432&vmname=srv-web1&node=proxmox4&c

```
  GNU nano 7.2                       /etc/hostname
srv-web1










                                   [ Read 1 line ]
^G Help         ^O Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location
^X Exit         ^R Read File    ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line
```

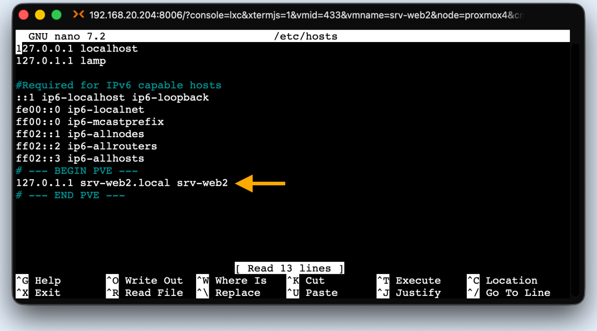

192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=432&vmname=srv-web1&node=proxmox4&c

```
  GNU nano 7.2                        /etc/hosts
127.0.0.1 localhost
127.0.1.1 lamp

#Required for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
# --- BEGIN PVE ---
127.0.1.1 srv-web1.local srv-web1
# --- END PVE ---




                                  [ Read 13 lines ]
^G Help         ^O Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location
^X Exit         ^R Read File    ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line
```

SRV-WEB2


192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=433&vmname=srv-web2&node=proxmox4&c

```
  GNU nano 7.2                    /etc/hostname
srv-web2

                              [ Read 1 line ]
^G Help          ^O Write Out     ^W Where Is     ^K Cut          ^T Execute      ^C Location
^X Exit          ^R Read File     ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line
```


192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=433&vmname=srv-web2&node=proxmox4&c

```
  GNU nano 7.2                     /etc/hosts
127.0.0.1 localhost
127.0.1.1 lamp

#Required for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
# --- BEGIN PVE ---
127.0.1.1 srv-web2.local srv-web2
# --- END PVE ---

                             [ Read 13 lines ]
^G Help          ^O Write Out     ^W Where Is     ^K Cut          ^T Execute      ^C Location
^X Exit          ^R Read File     ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line
```

1.2 - Installation d'Apache2 et du site web

Ensuite, installez **Apache2**.

```
root@srv-web1 /# apt update && apt upgrade -y

root@srv-web1 /# apt install apache2 php wget unzip -y
```

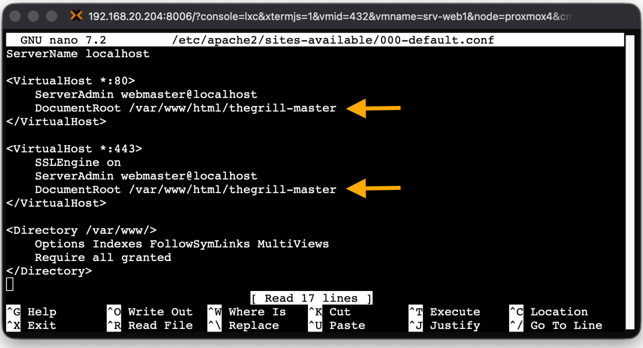
Enfin, rendez-vous sur le répertoire "**/var/www/html**" et installez un template de site comme celui-ci par exemple : https://github.com/technext/thegrill/archive/master.zip

```
root@srv-web1 /# cd /var/www/html

root@srv-web1 /# wget https://github.com/technext/thegrill/archive/master.zip

root@srv-web1 /# unzip master.zip
```

Après avoir extrait les fichiers compressés, allons modifier le fichier de configuration du site par défaut.

```
root@srv-web1 /# nano /etc/apache2/sites-available/000-default.conf
```

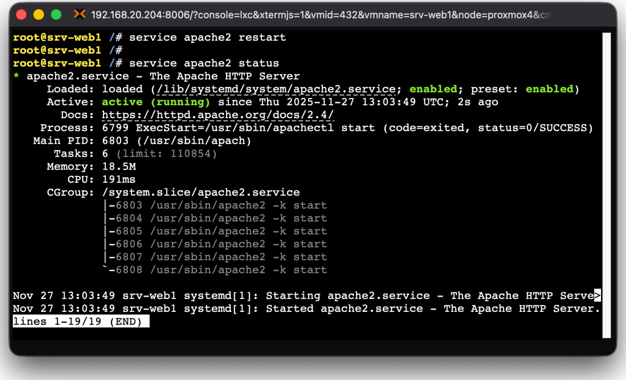
À l'intérieur de celui-ci, remplacez le chemin devant **DocumentRoot** pour pointer vers le répertoire de votre site.


192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=432&vmname=srv-web1&node=proxmox4&c

```
  GNU nano 7.2          /etc/apache2/sites-available/000-default.conf
ServerName localhost

<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/thegrill-master
</VirtualHost>

<VirtualHost *:443>
    SSLEngine on
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/thegrill-master
</VirtualHost>

<Directory /var/www/>
    Options Indexes FollowSymLinks MultiViews
    Require all granted
</Directory>

                                  [ Read 17 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line
```

Enfin, relançons et vérifions le statut de notre service **Apache**.

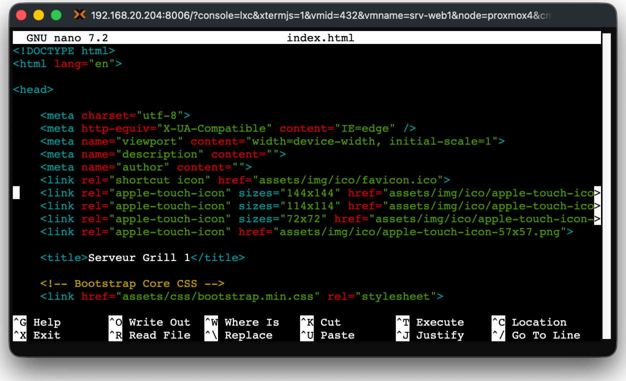
```
root@srv-web1 /# service apache2 restart

root@srv-web1 /# service apache2 status
```


192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=432&vmname=srv-web1&node=proxmox4&cr

```
root@srv-web1 /# service apache2 restart
root@srv-web1 /#
root@srv-web1 /# service apache2 status
* apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Thu 2025-11-27 13:03:49 UTC; 2s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 6799 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 6803 (/usr/sbin/apach)
      Tasks: 6 (limit: 110854)
     Memory: 18.5M
        CPU: 191ms
     CGroup: /system.slice/apache2.service
             |-6803 /usr/sbin/apache2 -k start
             |-6804 /usr/sbin/apache2 -k start
             |-6805 /usr/sbin/apache2 -k start
             |-6806 /usr/sbin/apache2 -k start
             |-6807 /usr/sbin/apache2 -k start
             `-6808 /usr/sbin/apache2 -k start

Nov 27 13:03:49 srv-web1 systemd[1]: Starting apache2.service - The Apache HTTP Serve>
Nov 27 13:03:49 srv-web1 systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-19/19 (END)
```

1.3 - Personnalisation de la page d'accueil

Allons modifier la page d'accueil (**index.html**) sur les deux serveurs en modifiant, par exemple, le titre de l'onglet ainsi que le titre de page.

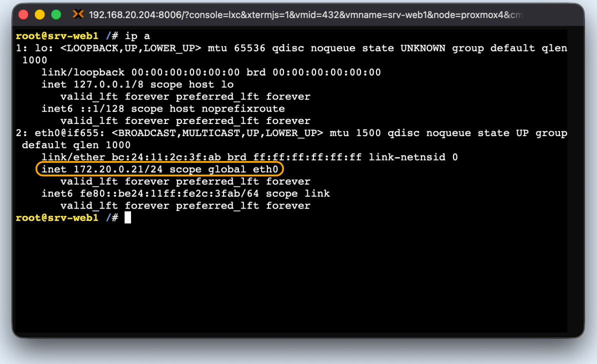

192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=432&vmname=srv-web1&node=proxmox4&cr

```
  GNU nano 7.2                          index.html
<!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-ico>
    <link rel="apple-touch-icon" sizes="114x114" href="assets/img/ico/apple-touch-ico>
    <link rel="apple-touch-icon" sizes="72x72" href="assets/img/ico/apple-touch-icon->
    <link rel="apple-touch-icon" href="assets/img/ico/apple-touch-icon-57x57.png">

    <title>Serveur Grill 1</title>

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

^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line
```

Étape 1 : Adressage IP des machines

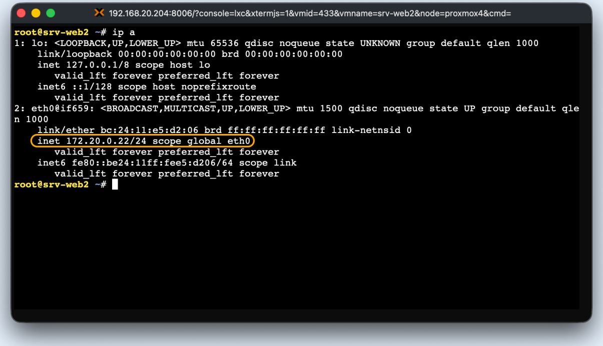
Adressons les machines comme indiqué dans la topologie du réseau.

SRV-WEB1


192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=432&vmname=srv-web1&node=proxmox4&cm

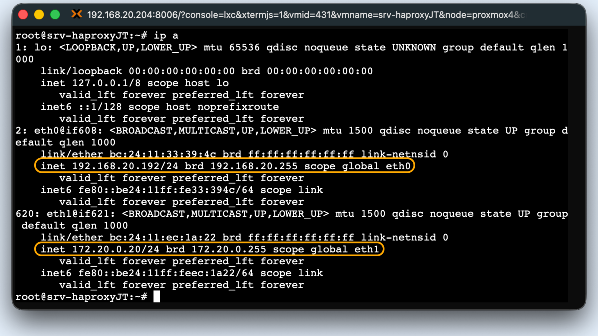
```
root@srv-web1 /# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen
 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0@if655: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group
 default qlen 1000
    link/ether bc:24:11:2c:3f:ab brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 172.20.0.21/24 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe2c:3fab/64 scope link
       valid_lft forever preferred_lft forever
root@srv-web1 /# 
```

SRV-WEB2


192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=433&vmname=srv-web2&node=proxmox4&cmd=

```
root@srv-web2 ~# ip a
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: eth0@if659: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qle
n 1000
    link/ether bc:24:11:e5:d2:06 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 172.20.0.22/24 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fee5:d206/64 scope link
       valid_lft forever preferred_lft forever
root@srv-web2 ~# 
```

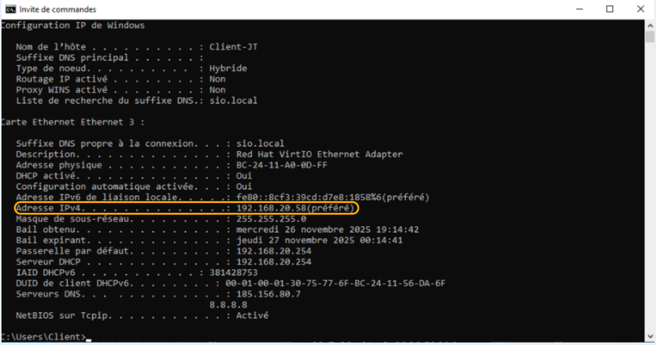
SRV-HAProxy



192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=431&vmname=srv-haproxyJT&node=proxmox4&c

```
root@srv-haproxyJT:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
000
    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: eth0@if608: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group d
efault qlen 1000
    link/ether bc:24:11:33:39:4c brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.168.20.192/24 brd 192.168.20.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe33:394c/64 scope link
       valid_lft forever preferred_lft forever
620: eth1@if621: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group
 default qlen 1000
    link/ether bc:24:11:ec:1a:22 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 172.20.0.20/24 brd 172.20.0.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:feec:1a22/64 scope link
       valid_lft forever preferred_lft forever
root@srv-haproxyJT:~# 
```

Client Windows



```
Configuration IP de Windows

   Nom de l’hôte . . . . . . . . . . : Client-JT
   Suffixe DNS principal . . . . . . :
   Type de noeud. . . . . . . . . .  : Hybride
   Routage IP activé . . . . . . . . : Non
   Proxy WINS activé . . . . . . . . : Non
   Liste de recherche du suffixe DNS.: sio.local

Carte Ethernet Ethernet 3 :

   Suffixe DNS propre à la connexion. . . : sio.local
   Description. . . . . . . . . . . . . . : Red Hat VirtIO Ethernet Adapter
   Adresse physique . . . . . . . . . . . : BC-24-11-A0-0D-FF
   DHCP activé. . . . . . . . . . . . . . : Oui
   Configuration automatique activée. . . : Oui
   Adresse IPv6 de liaison locale. . . . .: fe80::8cf3:39cd:d7e8:1858%6(préféré)
   Adresse IPv4. . . . . . . . . . . . . .: 192.168.20.58(préféré)
   Masque de sous-réseau. . . . . . . . . : 255.255.255.0
   Bail obtenu. . . . . . . . . . . . . . : mercredi 26 novembre 2025 19:14:42
   Bail expirant. . . . . . . . . . . . . : jeudi 27 novembre 2025 00:14:41
   Passerelle par défaut. . . . . . . . . : 192.168.20.254
   Serveur DHCP . . . . . . . . . . . . . : 192.168.20.254
   IAID DHCPv6 . . . . . . . . . . . : 381428753
   DUID de client DHCPv6. . . . . . . . : 00-01-00-01-30-75-77-6F-BC-24-11-56-DA-6F
   Serveurs DNS. . .  . . . . . . . . . . : 185.156.80.7
                                       8.8.8.8
   NetBIOS sur Tcpip. . . . . . . . . . . : Activé

C:\Users\Client>
```

Étape 2 : Installation et configuration de HAProxy

```
root@srv-web1 /# apt install haproxy -y
```

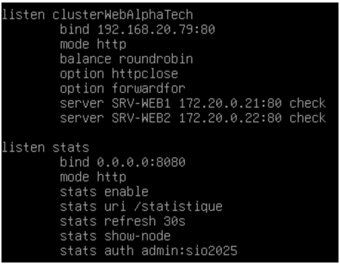
Puis, nous irons modifier le fichier de configuration principal :

```
root@srv-web1 /# nano /etc/haproxy/haproxy.cfg
```

À l'intérieur du fichier, rentrez les lignes suivantes :

```
listen clusterWebAlphaTech
    bind 192.168.0.20:80
    mode http
    balance roundrobin
    option httpclose
    option forwardfor
    server SRV-WEB1 172.20.0.21:80 check
    server SRV-WEB2 172.20.0.22:80 check

listen stats
    bind 0.0.0.0:8080
    mode http
    stats enable
    stats uri /statistique
    stats refresh 30s
    stats show-node
    stats auth admin:password
```

Ce qui devrait donner ceci :

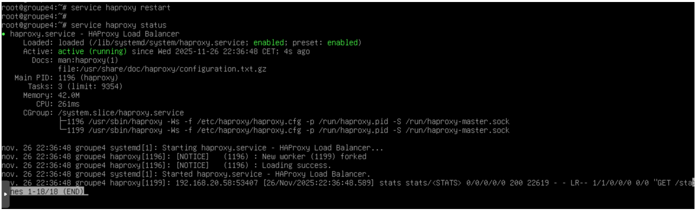


```
listen clusterWebAlphaTech
        bind 192.168.20.79:80
        mode http
        balance roundrobin
        option httpclose
        option forwardfor
        server SRV-WEB1 172.20.0.21:80 check
        server SRV-WEB2 172.20.0.22:80 check

listen stats
        bind 0.0.0.0:8080
        mode http
        stats enable
        stats uri /statistique
        stats refresh 30s
        stats show-node
        stats auth admin:sio2025
```

Ensuite, redémarrons le service HAProxy :

```
root@srv-web1 /# service haproxy restart
```

```
root@srv-web1 /# service haproxy status
```



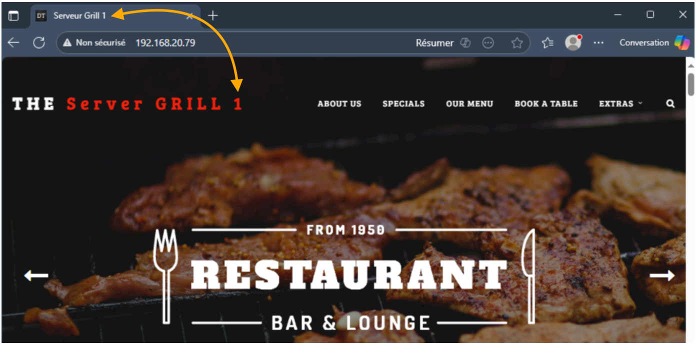
```
root@groupe4:~# service haproxy restart
root@groupe4:~#
root@groupe4:~# service haproxy status
• haproxy.service - HAProxy Load Balancer
     Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-11-26 22:36:48 CET; 4s ago
       Docs: man:haproxy(1)
             file:/usr/share/doc/haproxy/configuration.txt.gz
   Main PID: 1196 (haproxy)
      Tasks: 3 (limit: 9354)
     Memory: 42.0M
        CPU: 261ms
     CGroup: /system.slice/haproxy.service
             ├─1196 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
             └─1199 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

nov. 26 22:36:48 groupe4 systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
nov. 26 22:36:48 groupe4 haproxy[1196]: [NOTICE]   (1196) : New worker (1199) forked
nov. 26 22:36:48 groupe4 haproxy[1196]: [NOTICE]   (1196) : Loading success.
nov. 26 22:36:48 groupe4 systemd[1]: Started haproxy.service - HAProxy Load Balancer.
 v. 26 22:36:48 groupe4 haproxy[1199]: 192.168.20.58:53407 [26/Nov/2025:22:36:48.589] stats stats/<STATS> 0/0/0/0/0 200 22619 - - LR-- 1/1/0/0/0 0/0 "GET /sta
 nes 1-18/18 (END)
```

HAProxy est désormais installé et actif, il ne reste plus qu'à essayer côté client... ✅

Étape 3 : Validation et tests

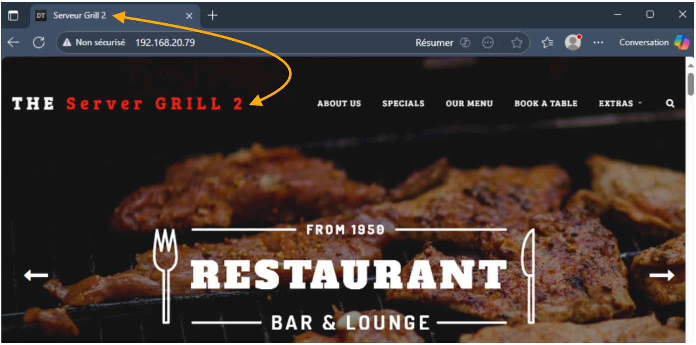
1.1 - Test de la répartition de charge

Sur votre navigateur, rentrez l'adresse IP de votre serveur HAProxy (ici : **192.168.20.79**).



Comme nous pouvons le constater, il renvoie actuellement sur le premier serveur web (**172.20.0.21**).

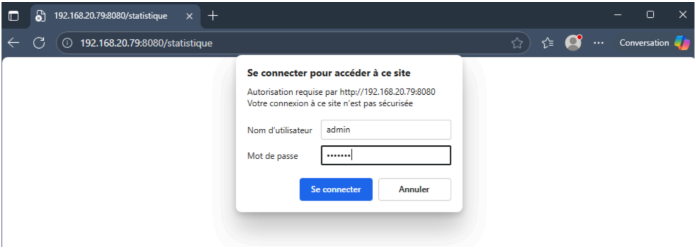
Maintenant, tentons de rafraîchir la page...



Comme nous pouvons le remarquer, le site bascule vers le second serveur. ✅
Voici comment fonctionne en cas pratique un équilibreur de charge.

1.2 - Supervision du cluster

Nous pouvons accéder à un tableau de bord de statistique qui permet de visualiser l'état du load balancer (l'équilibreur de charge) et des serveurs.
Dans le cadre de ce TP, nous pouvons y accéder à l'adresse suivante : http://192.168.20.79:8080/statistique.

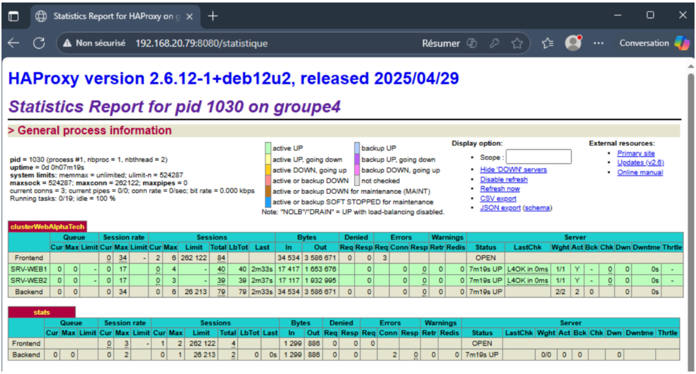

192.168.20.79:8080/statistique
192.168.20.79:8080/statistique
Conversation
Se connecter
Annuler

Se connecter pour accéder à ce site

Autorisation requise par http://192.168.20.79:8080
Votre connexion à ce site n'est pas sécurisée

Nom d'utilisateur: admin
Mot de passe: •••••••

Sur cette page, nous pouvons voir l'état de nos serveurs (**UP** ou **DOWN**), surveiller la charge et le trafic, permettre de diagnostiquer des problèmes (pannes, saturations, connexions anormales) et de comprendre comment HAProxy distribue les requêtes.


Statistics Report for HAProxy on g
Non sécurisé 192.168.20.79:8080/statistique
Résumer
Conversation

HAProxy version 2.6.12-1+deb12u2, released 2025/04/29

Statistics Report for pid 1030 on groupe4

> General process information

pid = 1030 (process #1, nbproc = 1, nbthread = 2)
uptime = 0d 0h07m19s
system limits: memmax = unlimited; ulimit-n = 524287
maxsock = 524287; **maxconn** = 262122; **maxpipes** = 0
current conns = 3; current pipes = 0/0; conn rate = 0/sec; bit rate = 0.000 kbps
Running tasks: 0/19; idle = 100 %

active UP
active UP, going down
active DOWN, going up
active or backup DOWN
active or backup DOWN for maintenance (MAINT)
active or backup SOFT STOPPED for maintenance
backup UP
backup UP, going down
backup DOWN, going up
not checked
Note: "NOLB"/"DRAIN" = UP with load-balancing disabled.

Display option:
- Scope :
- Hide 'DOWN' servers
- Disable refresh
- Refresh now
- CSV export
- JSON export (schema)

External resources:
- Primary site
- Updates (v2.6)
- Online manual

clusterWebAlphaTech

	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				0	34	-	2	6	262 122	84			34 534	3 586 671	0	0	3					OPEN								
SRV-WEB1	0	0	-	0	17		0	4	-	40	40	2m33s	17 417	1 653 676		0		0	0	0	0	7m19s UP	L4OK in 0ms	1/1	Y	-	0	0	0s	-
SRV-WEB2	0	0	-	0	17		0	3	-	39	39	2m37s	17 117	1 932 995		0		0	0	0	0	7m19s UP	L4OK in 0ms	1/1	Y	-	0	0	0s	-
Backend	0	0		0	34		0	6	26 213	79	79	2m33s	34 534	3 586 671	0	0		0	0	0	0	7m19s UP		2/2	2	0		0	0s	

stats

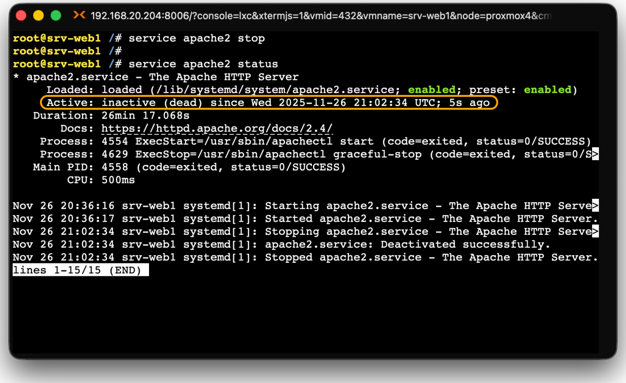
	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				0	3	-	1	2	262 122	4			1 299	886	0	0	0					OPEN								
Backend	0	0		0	2		0	1	26 213	2	0	0s	1 299	886	0	0		2	0	0	0	7m19s UP		0/0	0	0		0		

Frontend : représente les points d'entrée du trafic. Il est utile pour savoir si HAProxy reçoit du trafic et à quel rythme.

Backend : représente le groupe de serveurs derrière un frontend. Il permet de savoir si l'ensemble du cluster répond correctement.

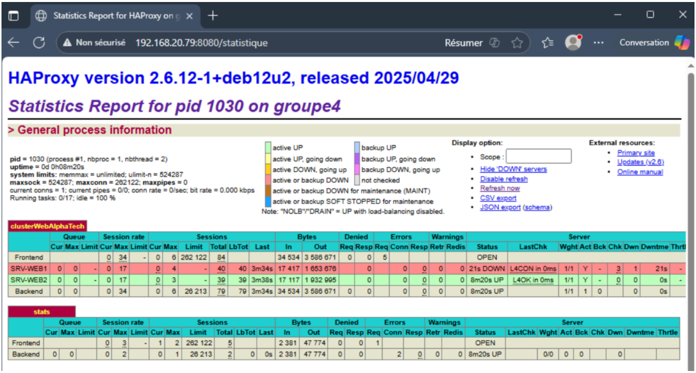
1.3 - Test de tolérance de panne

Arrêtons le service Apache sur le serveur web 1 par exemple :


192.168.20.204:8006/?console=lxc&xtermjs=1&vmid=432&vmname=srv-web1&node=proxmox4&cm

```
root@srv-web1 /# service apache2 stop
root@srv-web1 /#
root@srv-web1 /# service apache2 status
* apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: inactive (dead) since Wed 2025-11-26 21:02:34 UTC; 5s ago
   Duration: 26min 17.068s
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 4554 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
    Process: 4629 ExecStop=/usr/sbin/apachectl graceful-stop (code=exited, status=0/S>
   Main PID: 4558 (code=exited, status=0/SUCCESS)
        CPU: 500ms

Nov 26 20:36:16 srv-web1 systemd[1]: Starting apache2.service - The Apache HTTP Serve>
Nov 26 20:36:17 srv-web1 systemd[1]: Started apache2.service - The Apache HTTP Server.
Nov 26 21:02:34 srv-web1 systemd[1]: Stopping apache2.service - The Apache HTTP Serve>
Nov 26 21:02:34 srv-web1 systemd[1]: apache2.service: Deactivated successfully.
Nov 26 21:02:34 srv-web1 systemd[1]: Stopped apache2.service - The Apache HTTP Server.
lines 1-15/15 (END)
```

Attendez 30 secondes.
Ensuite, retournez sur votre tableau de bord et vous devriez voir **SRV-WEB1** down et en rouge.


Statistics Report for HAProxy on g
Non sécurisé
192.168.20.79:8080/statistique
Résumer
Conversation

HAProxy version 2.6.12-1+deb12u2, released 2025/04/29

Statistics Report for pid 1030 on groupe4

> General process information

pid = 1030 (process #1, nbproc = 1, nbthread = 2)
uptime = 0d 0h08m20s
system limits: memmax = unlimited; ulimit-n = 524287
maxsock = 524287; **maxconn** = 262122; **maxpipes** = 0
current conns = 1; current pipes = 0/0; conn rate = 0/sec; bit rate = 0.000 kbps
Running tasks: 0/17; idle = 100 %

- active UP
- active UP, going down
- active DOWN, going up
- active or backup DOWN
- active or backup DOWN for maintenance (MAINT)
- active or backup SOFT STOPPED for maintenance
- backup UP
- backup UP, going down
- backup DOWN, going up
- not checked

Note: "NOLB"/"DRAIN" = UP with load-balancing disabled.

Display option:
- Scope :
- Hide 'DOWN' servers
- Disable refresh
- Refresh now
- CSV export
- JSON export (schema)

External resources:
- Primary site
- Updates (v2.6)
- Online manual

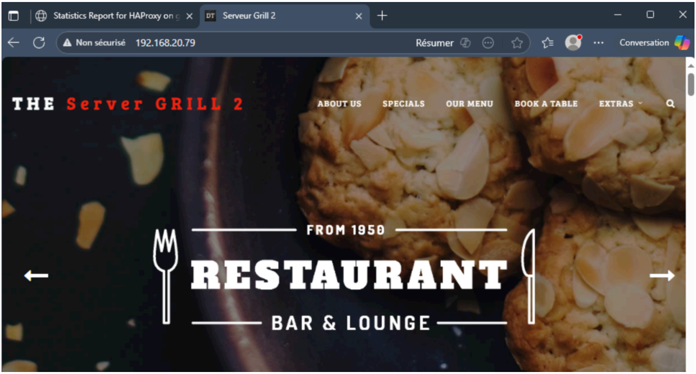
clusterWebAlphaTech

	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server							
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtl
Frontend				0	34	-	0	6	262 122	84			34 534	3 586 671	0	0	5					OPEN								
SRV-WEB1	0	0	-	0	17		0	4	-	40	40	3m34s	17 417	1 653 676		0		0	0	0	0	21s DOWN	L4CON in 0ms	1/1	Y	-	3	1	21s	-
SRV-WEB2	0	0	-	0	17		0	3	-	39	39	3m38s	17 117	1 932 995		0		0	0	0	0	8m20s UP	L4OK in 0ms	1/1	Y	-	0	0	0s	-
Backend	0	0		0	34		0	6	26 213	79	79	3m34s	34 534	3 586 671	0	0		0	0	0	0	8m20s UP		1/1	1	0		0	0s	

stats

	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server							
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				0	3	-	1	2	262 122	5			2 381	47 774	0	0	1					OPEN								
Backend	0	0		0	2		0	1	26 213	2	0	0s	2 381	47 774	0	0		2	0	0	0	8m20s UP		0/0	0	0		0		

Et, si nous retournons sur le site une nouvelle fois, sans surprise, nous sommes envoyé sur le serveur web 2.



En conclusion...

Ce TP a permis de démontrer l'intérêt d'une architecture de répartition de charge avec HAProxy pour améliorer la **disponibilité** d'un site web.

La configuration du load balancer, l'intégration des serveurs web et les tests de bascule ont montré que la continuité de service peut être assurée même en cas de panne d'un des serveurs.

TRIOUX Jimmy - BTS SIO2 SISR