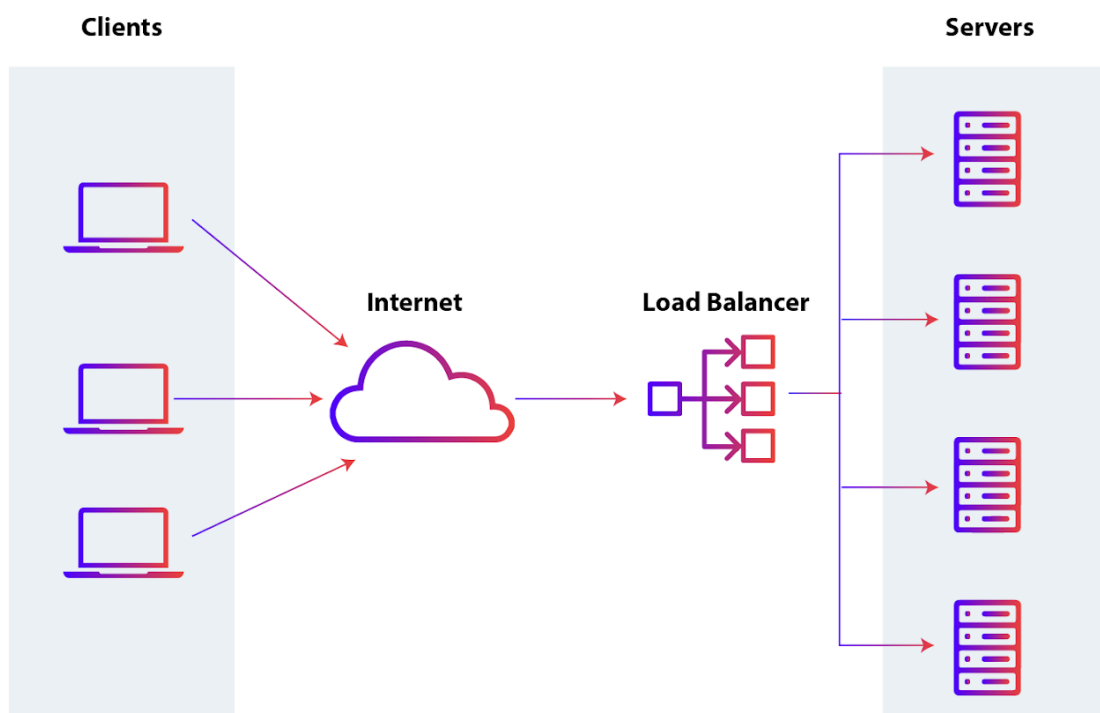


15/10/2024

Jourdan Alix  
Savic César

BTS SIO2

## TP 3 : Mise en place d'une solution de répartition des services avec IPVS et un Load Balancer



## Sommaire :

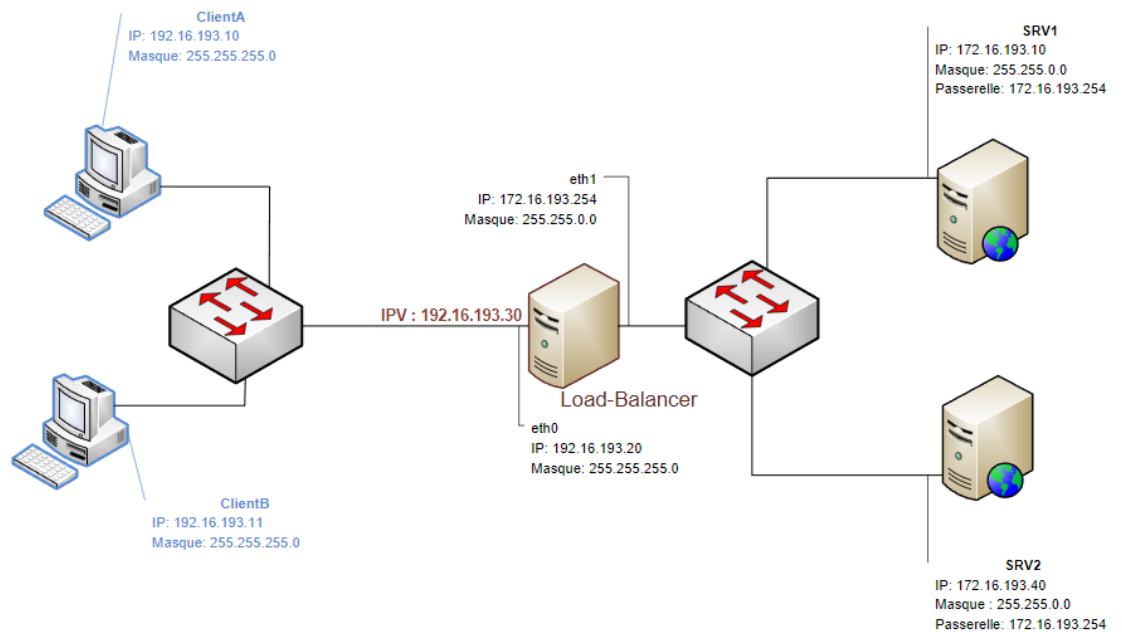
|                                                |          |
|------------------------------------------------|----------|
| Sommaire :.....                                | 2        |
| Objectif du Load-Balancer :.....               | 3        |
| <b>Architecture du réseau :.....</b>           | <b>3</b> |
| Création des serveur :.....                    | 4        |
| Création du répartiteur (Load-Balancer) :..... | 7        |
| Test du répartiteur de charge :.....           | 11       |
| conclusion :.....                              | 12       |

## Objectif du Load-Balancer :



Un load balancer est un service utilisé dans l'infrastructure réseau pour agir comme un répartiteur de charge. Son objectif principal est de gérer les activités liées aux requêtes envoyées vers les serveurs. Il utilise des algorithmes, comme le *Round Robin*, pour répartir les requêtes de manière équitable entre les serveurs, afin d'éviter qu'un seul serveur soit surchargé et garantir une meilleure performance et disponibilité des services.

## Architecture du réseau :



## Création des serveur :

Tout d'abord, nous allons installer les serveurs sur notre ESXi :

**Installation de Debian 12:** Sur le serveur ESXi, nous avons créé les deux serveurs qui feront partie du cluster.



**Installation d'Apache2:** Pour installer Apache2 sur les serveurs, nous devons exécuter la commande suivante :

```
root@debian:/home/ # apt install apache2
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
apache2 est déjà la version la plus récente (2.4.62-1~deb12u2).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 17 non mis à jour.
```



## Apache2 Ubuntu Default Page

ubuntu

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

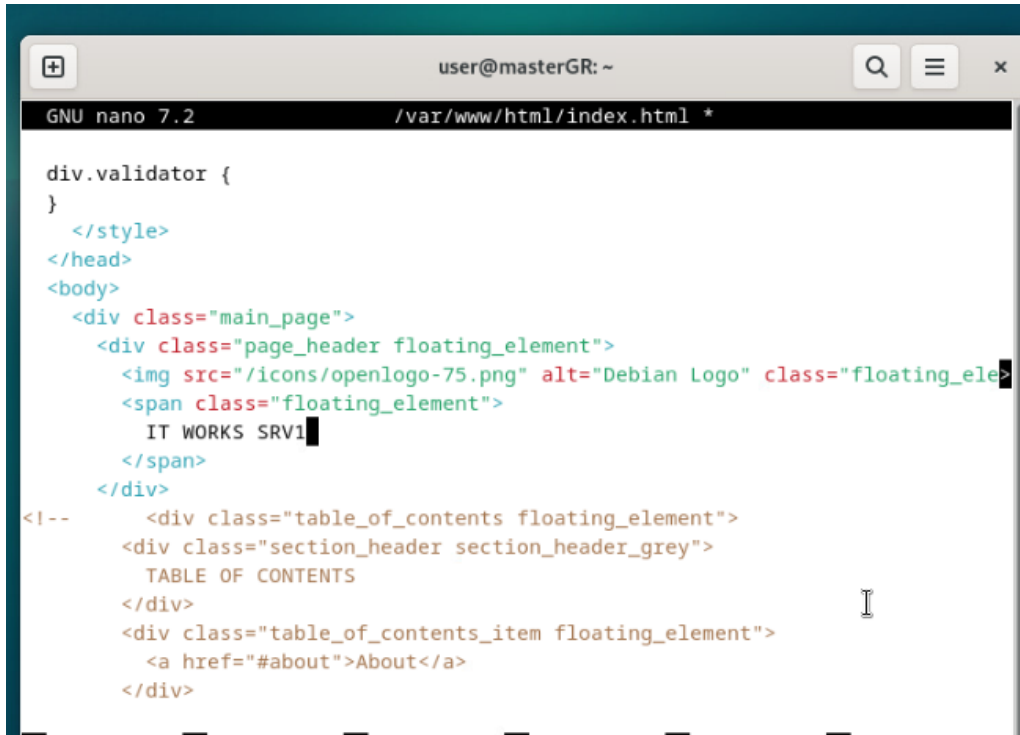
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.so
```

modification du titre la page web :

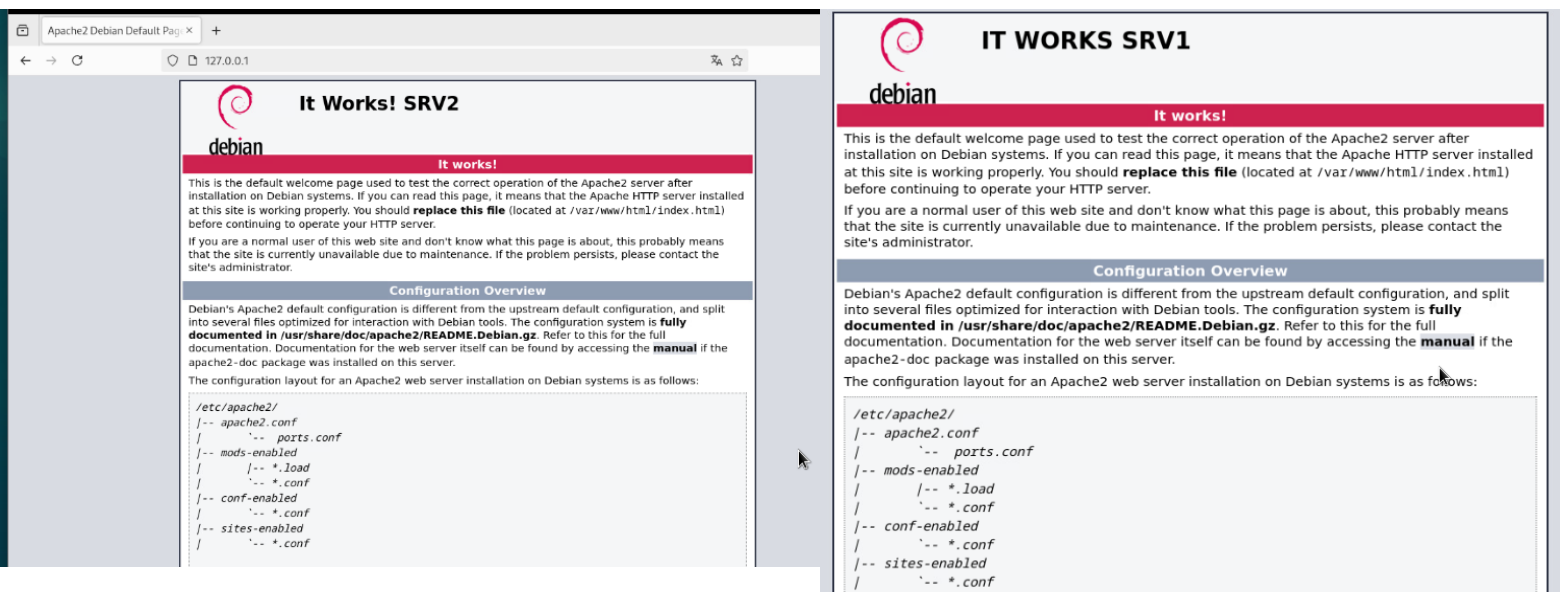
avec la commande nano /var/www/html/index.html



```
GNU nano 7.2 /var/www/html/index.html *

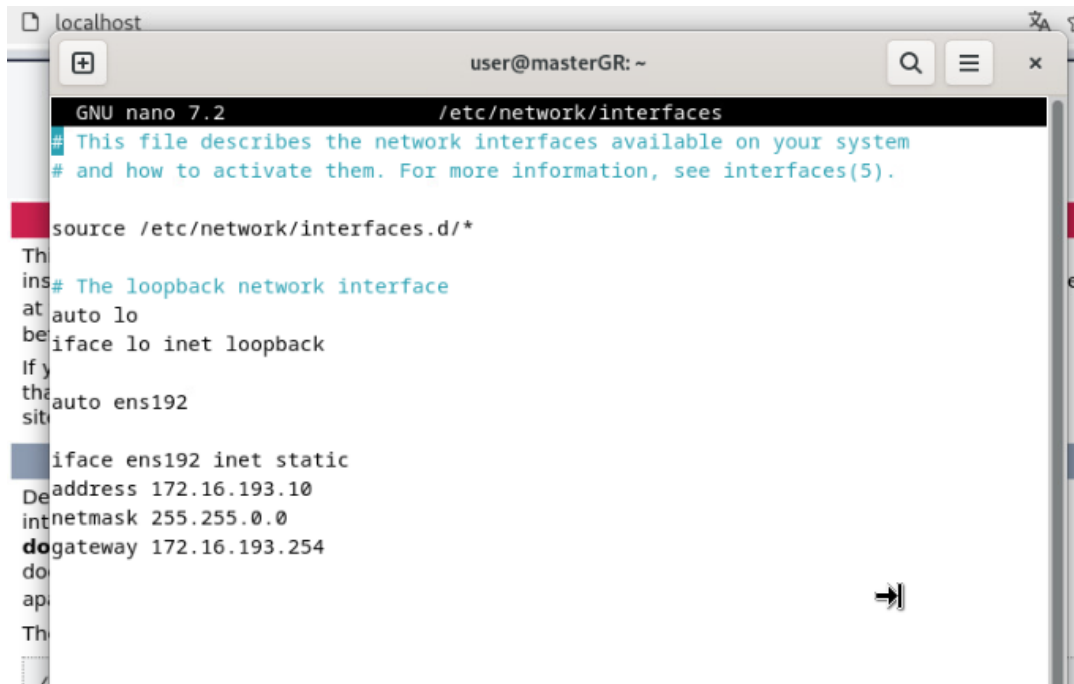
div.validator {
}
</style>
</head>
<body>
  <div class="main_page">
    <div class="page_header floating_element">
      
        IT WORKS SRV1
      </span>
    </div>
  <!--
    <div class="table_of_contents floating_element">
      <div class="section_header section_header_grey">
        TABLE OF CONTENTS
      </div>
      <div class="table_of_contents_item floating_element">
        <a href="#about">About</a>
      </div>
```

Vérification de l'affichage après modification du titre de la page :



## Configuration de l'adresse IP définitive sur chacun des deux serveurs :

plan d'adressage serveur 1 :



```

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).

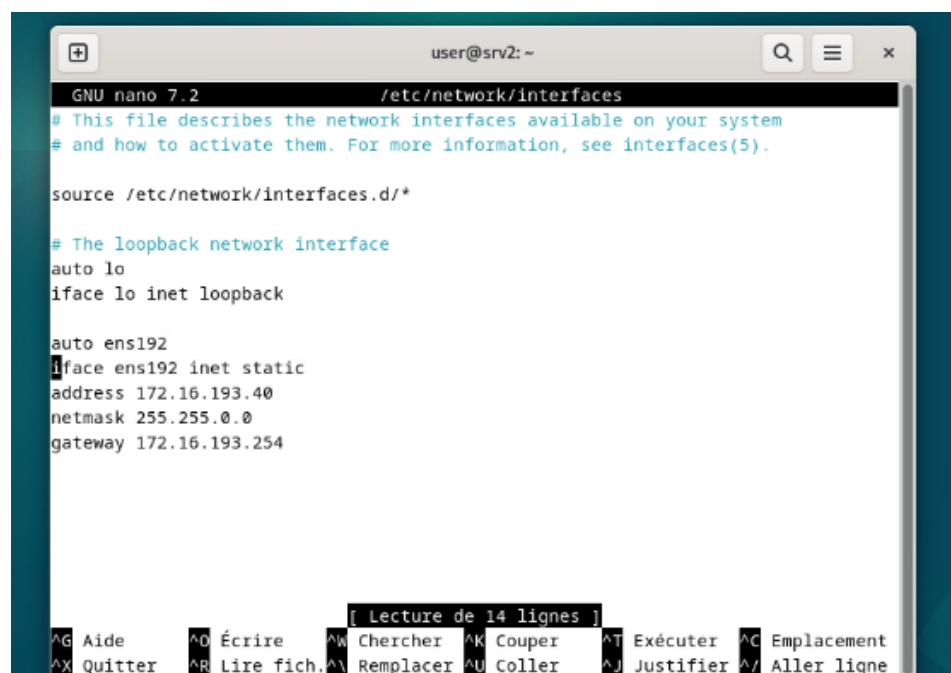
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto ens192
iface ens192 inet static
address 172.16.193.10
netmask 255.255.0.0
gateway 172.16.193.254

```

plan d'adressage serveur 2 :



```

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

auto ens192
iface ens192 inet static
address 172.16.193.40
netmask 255.255.0.0
gateway 172.16.193.254

```

[ Lecture de 14 lignes ]

^G Aide    ^O Écrire    ^W Chercher    ^K Couper    ^T Exécuter    ^C Emplacement  
 ^X Quitter    ^R Lire fich.    ^U Remplacer    ^J Coller    ^J Justifier    ^\_ Aller ligne

Séparation des serveurs du réseau principal du lycée :

Création de switch virtuel :

switch TP.3 cybersécurité AC

## Création du répartiteur (Load-Balancer) :

Pour la deuxième partie, nous allons éteindre les serveurs afin d'éviter tout conflit d'adresse IP, puis créer une nouvelle machine Linux qui servira de répartiteur :

serveur Load-Balancer :

et

ajouter une seconde carte réseaux :  
aller dans les paramètres et sélectionner



installation du package Ipv6 sur le répartiteur :  
avec la commande apt install ipvsadm

```
274 paquets peuvent être mis à jour. exécutez « apt list --upgradable » pour les voir.
root@debian:/home/administrateur# apt-get install ipvsadm
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Paquets suggérés :
  heartbeat keepalived ldirectord
Les NOUVEAUX paquets suivants seront installés :
  ipvsadm
0 mis à jour, 1 nouvellement installés, 0 à enlever et 274 non mis à jour.
Il est nécessaire de prendre 41,6 ko dans les archives.
Après cette opération, 131 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 ipvsadm amd64 1:1.31-1+b1 [41,6 kB]
41,6 ko réceptionnés en 0s (657 ko/s)
Sélection du paquet ipvsadm précédemment désélectionné.
(Lecture de la base de données... 149838 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../ipvsadm_1%3a1.31-1+b1_amd64.deb ...
Dépaquetage de ipvsadm (1:1.31-1+b1) ...
Paramétrage de ipvsadm (1:1.31-1+b1) ...
Traitement des actions différées (« triggers ») pour man-db (2.11.2-2) ...
root@debian:/home/administrateur#
```

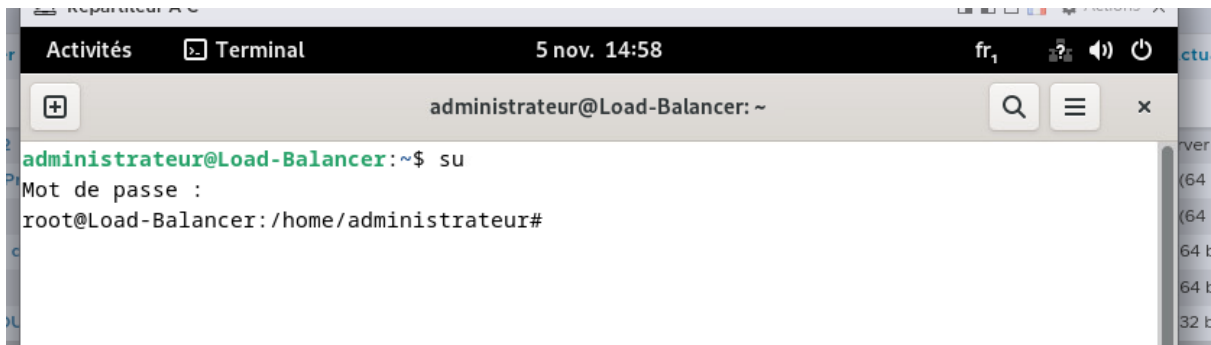
Attribution des deux address Ip :

```

2: ens192: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
    link/ether 00:0c:29:0d:23:14 brd ff:ff:ff:ff:ff:ff
    altname enp11s0
    inet 172.16.193.254/16 brd 172.16.255.255 scope global ens192
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe0d:2314/64 scope link
        valid_lft forever preferred_lft forever
3: ens224: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
    link/ether 00:0c:29:0d:23:1e brd ff:ff:ff:ff:ff:ff
    altname enp19s0
    inet 192.16.193.20/16 brd 192.16.255.255 scope global ens224
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe0d:231e/64 scope link
        valid_lft forever preferred_lft forever
administrateur@debian:~$

```

Renommer le serveur :  
avec la commande nano /etc/hostname



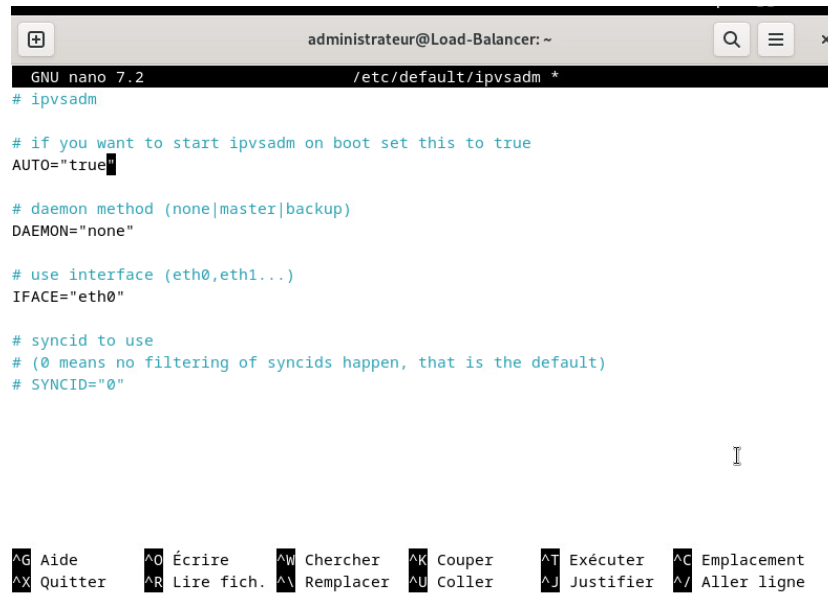
```

Administrateur 5 nov. 14:58 fr_1 ctu
+ administrateur@Load-Balancer: ~
administrateur@Load-Balancer:~$ su
Mot de passe :
root@Load-Balancer: /home/administrateur#

```

Editer le fichier de configuration de IPVS :

Configurez le fichier `/etc/default/ipvsadm` pour activer le chargement des règles IPVS au démarrage en définissant l'option sur `true`, ce qui démarre directement le service. Si vous ne souhaitez pas charger de règles, indiquez `none`.



```

administrateur@Load-Balancer: ~
GNU nano 7.2 /etc/default/ipvsadm *
# ipvsadm

# if you want to start ipvsadm on boot set this to true
AUTO="true"

# daemon method (none|master|backup)
DAEMON="none"

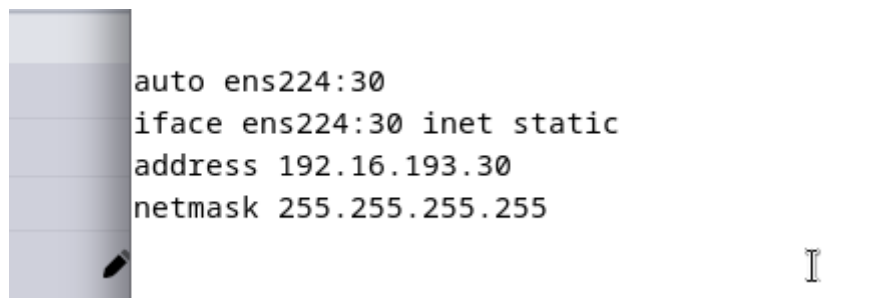
# use interface (eth0,eth1...)
IFACE="eth0"

# syncid to use
# (0 means no filtering of syncids happen, that is the default)
# SYNCID="0"

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^_/ Aller ligne
  
```

Création de la carte virtuelle :

Nous allons ajouter une carte virtuelle qui sera partagée afin que tous les services soient accessibles aux utilisateurs. Cette carte virtuelle sera configurée comme passerelle (gateway) pour le load balancer. La configuration se fera dans le fichier `/etc/network/interfaces` à l'aide de l'éditeur Nano



```

auto ens224:30
iface ens224:30 inet static
address 192.16.193.30
netmask 255.255.255.255
  
```

L'interface `ens224:30` représente une carte réseau virtuelle basée sur la carte physique principale `ens224`. Le numéro `:30` indique qu'il s'agit d'une configuration supplémentaire créée pour ajouter une nouvelle adresse IP ou permettre des connexions spécifiques.

```

2: ens192: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
    link/ether 00:0c:29:0d:23:14 brd ff:ff:ff:ff:ff:ff
    altname enp11s0
    inet 172.16.193.254/16 brd 172.16.255.255 scope global ens192
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe0d:2314/64 scope link
        valid_lft forever preferred_lft forever
3: ens224: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
    link/ether 00:0c:29:0d:23:1e brd ff:ff:ff:ff:ff:ff
    altname enp19s0
    inet 192.16.193.20/16 brd 192.16.255.255 scope global ens224
        valid_lft forever preferred_lft forever
    inet 192.16.193.30/32 brd 192.16.193.30 scope global ens224:30
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe0d:231e/64 scope link
        valid_lft forever preferred_lft forever
administrateur@Load-Balancer:~$

```

maintenant nous allons activer le routage avec la commande suivante :

```

root@Load-Balancer:/home/administrateur# echo 1 > /proc/sys/net/ipv4/ip_forward
root@Load-Balancer:/home/administrateur#

```

paramétrage du cluster :

cette commande sert à dire aux deux serveurs que la carte virtuelle existe

```

ipvsadm --add-service -t 192.16.193.30:http -s rr
ipvsadm -a -t 192.16.193.30:http -r 172.16.193.10:http -m -w 1
ipvsadm -a -t 192.16.193.30:http -r 172.16.193.40:http -m -w 1

```

commande pour vérifier le paramétrage :

```

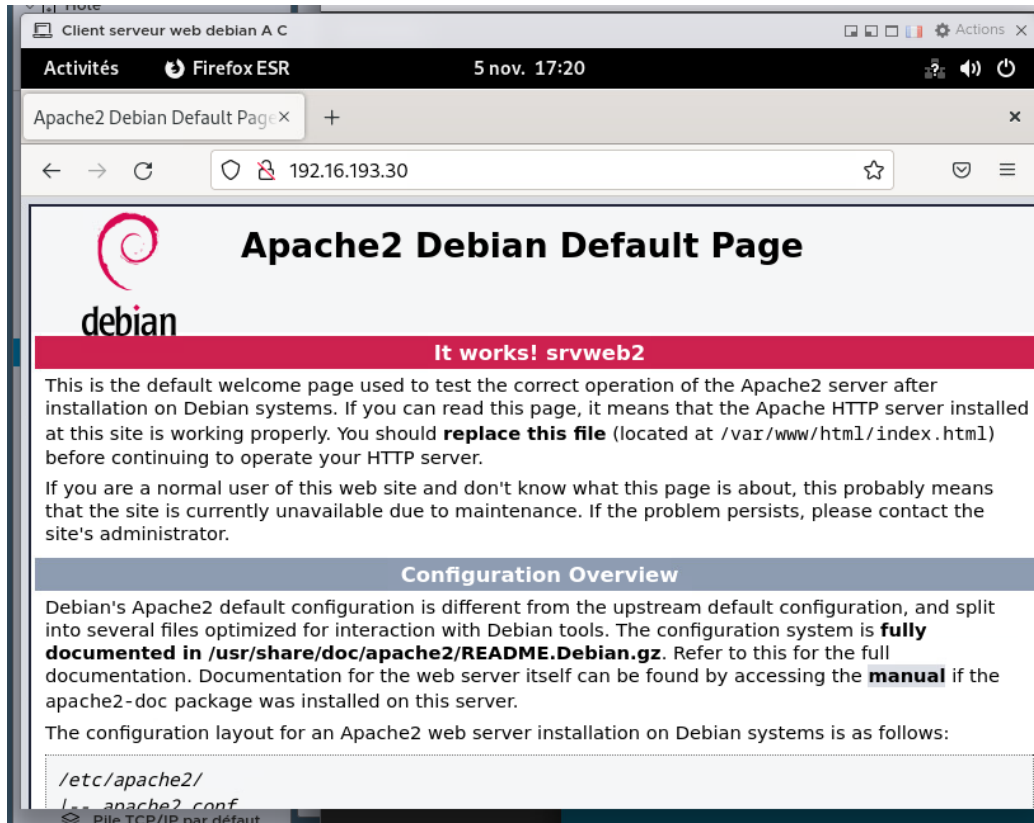
root@Load-Balancer:/home/administrateur# sudo ipvsadm -L
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
TCP  192.16.193.10:http rr
TCP  Load-Balancer:http rr
  -> 172.16.193.10:http             Masq    1      0      0
  -> 172.16.193.40:http             Masq    1      0      0
root@Load-Balancer:/home/administrateur# sudo ipvsadm -Ln
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port           Forward Weight ActiveConn InActConn
TCP  192.16.193.10:80 rr
TCP  192.16.193.30:80 rr
  -> 172.16.193.10:80               Masq    1      0      0
  -> 172.16.193.40:80               Masq    1      0      0

```

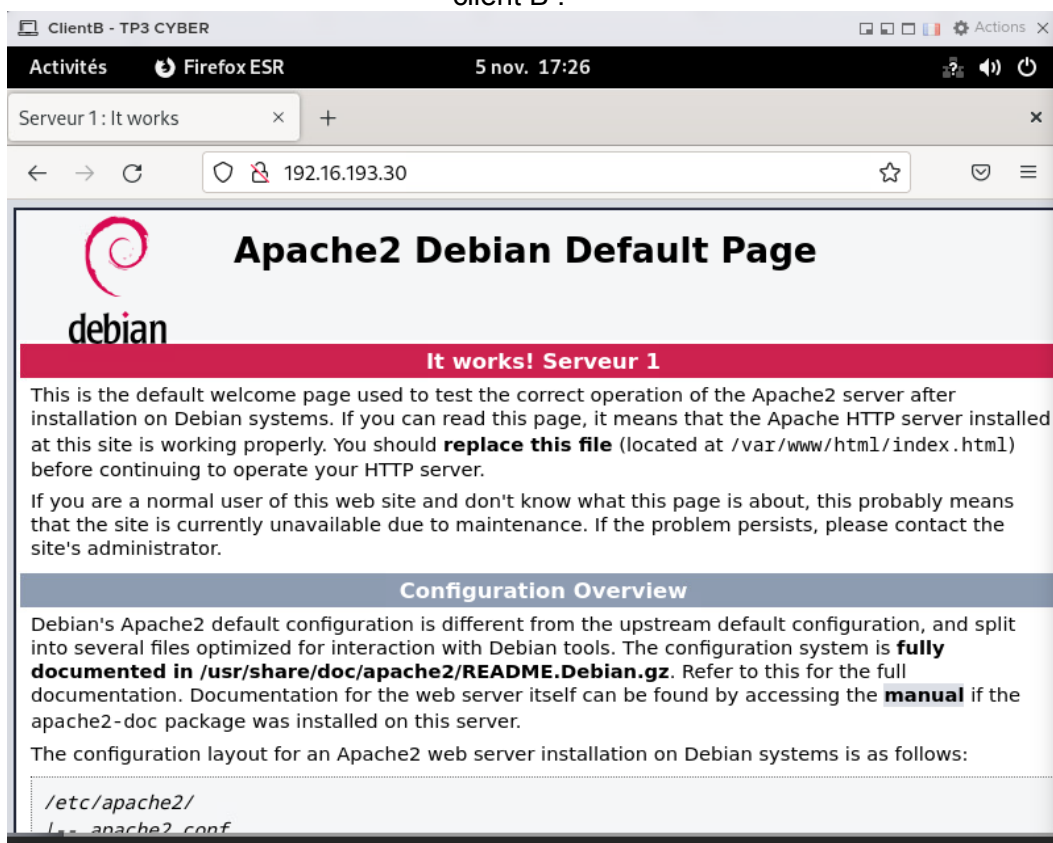
## Test du répartiteur de charge :

Dans la deuxième partie, nous allons tester pour vérifier si les utilisateurs accèdent correctement au serveur et si le répartiteur de charge fonctionne correctement. Nous avons également configuré les machines clientes conformément au schéma.

client A :



client B :



affiche les connexions actives ou inactives

```

root@Load-Balancer:/home/repartiteur1# sudo ipvsadm -L
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port          Forward Weight ActiveConn InActConn
TCP  Load-Balancer:http rr
  -> 172.16.193.10:http           Masq    1      0          0
  -> 172.16.193.40:http           Masq    1      0          0

```

## conclusion :

Grâce à la mise en place de cette architecture réseau, incluant un Load Balancer configuré avec IPVS, nous avons réussi à répartir efficacement les requêtes entre les serveurs du cluster. Les tests réalisés montrent que :

1. Les utilisateurs accèdent correctement aux serveurs via la carte virtuelle, utilisée comme passerelle pour le Load Balancer.
2. Le répartiteur de charge distribue les requêtes de manière équilibrée, évitant ainsi la surcharge d'un seul serveur.
3. Les machines clientes, configurées selon le plan d'adressage, interagissent correctement avec le réseau, confirmant la fiabilité et la stabilité de notre configuration.

Cette solution garantit une meilleure disponibilité, une optimisation des ressources et une architecture évolutive, adaptée aux besoins des utilisateurs et des services critiques.