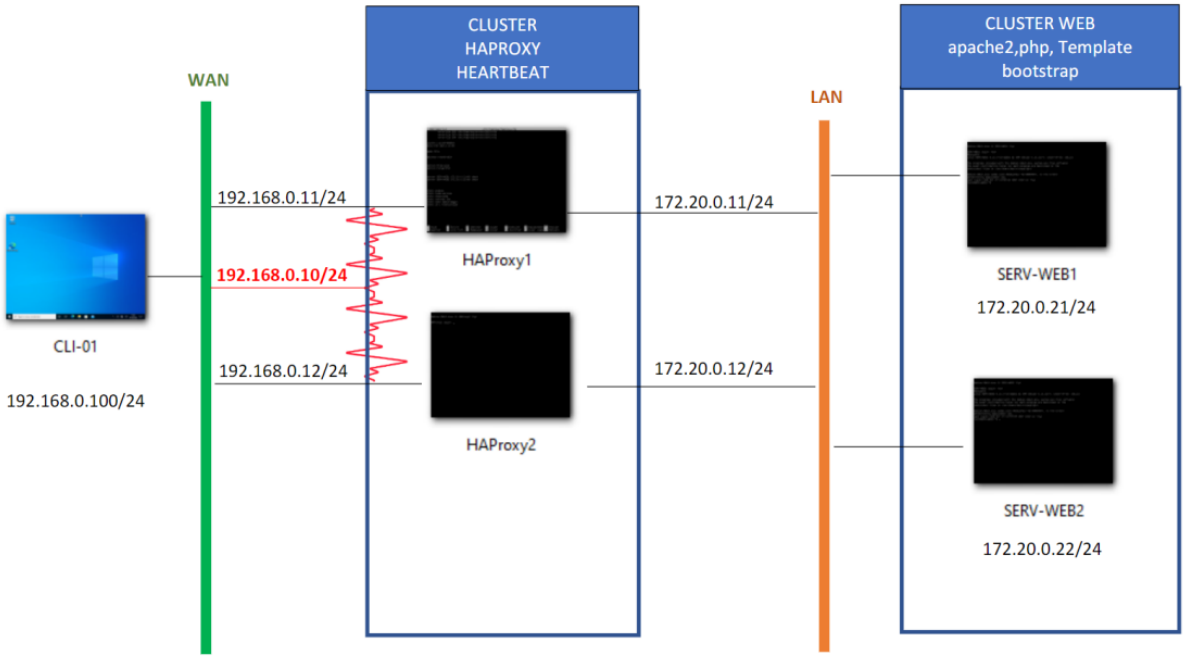
Système et Réseau

Livrable HIGH AVAILABILITY : HAProxy & Hearbeat Sous Debian 12

Réseau à créer :



Chloé Houivet & Kévin Roussel BTS SIO 1LM

Sommaire : Installation des machines virtuelles, configuration des adresses IP et de HAProxy et Heartbeat

Les étapes ainsi que les captures d'écrans sont numérotées.

Analyse de la demande

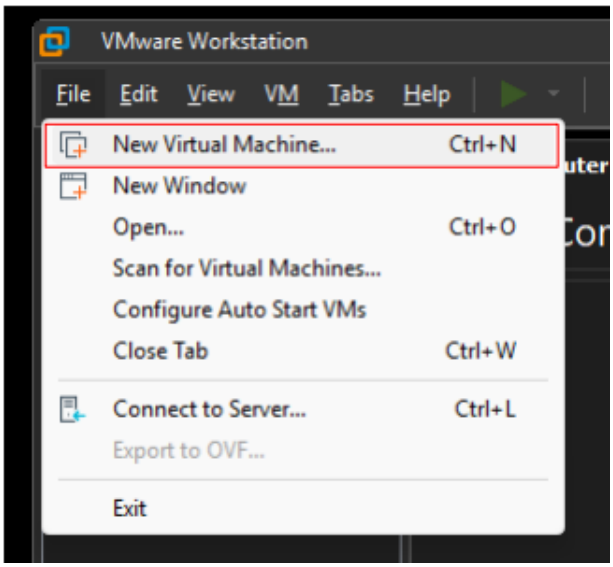
Dans le schéma du réseau, l'on souhaite avoir 5 machines

Machine	OS	Caractéristiques
Client Windows	Windows 10	Ip statique 192.168.0.100 NAT en dhcp
HAProxy 1	Debian	IP statique 172.20.0.11/24 IP statique 192.168.0.11/24 NAT en dhcp Heartbeat en 192.168.0.10/24
HAProxy 2	Debian	IP statique 172.20.0.12/24 IP statique 192.168.0.12/24 NAT en dhcp Heartbeat en 192.168.0.10/24
SRV-WEB 1	Debian	IP statique 172.20.0.21/24
SRV-WEB 2	Debian	IP statique 172.20.0.22/24

Segment LAN	Machines connectées
Segment LAN 1	Connecte HAProxy1, HAProxy2, SRV-WEB1 et SRV-WEB2 sur le même réseau (172.20.0.0/24)
Segment LAN 2	Connecte HAProxy1, HAProxy2 et Client Windows sur le même réseau (192.168.0.0/24)

1. ***Procédure de l'installation des machines***
    a. ***Installation de la machine virtuelle DEBIAN sans interface graphique (SRV-WEB1)***

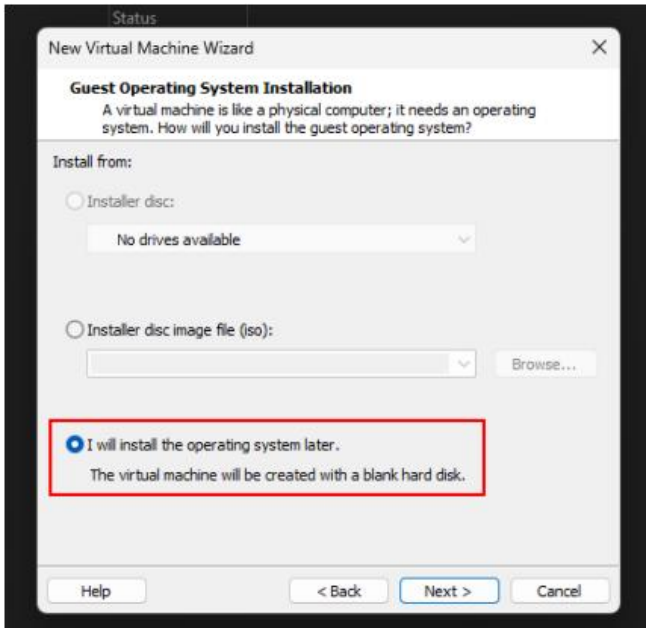
Etape 1 :



1. Appuyer sur file puis sur « New Virtual Machine… ».

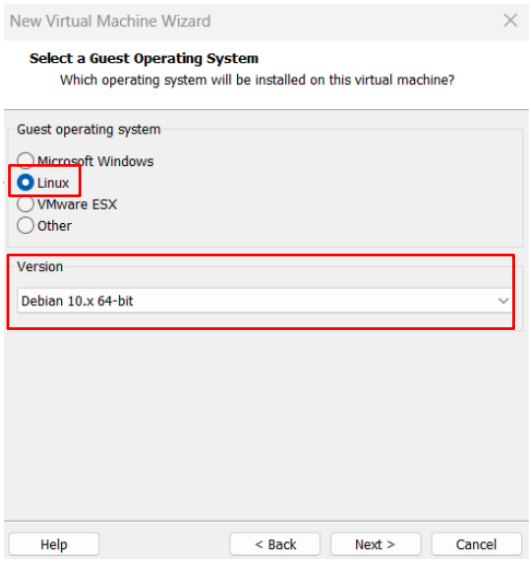
2. Choisir le type de la machine « Typical (recommanded) ».

Etape 2 :



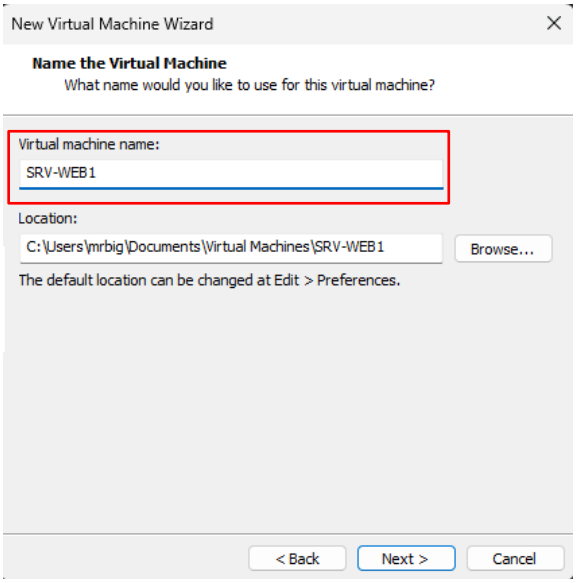
Choisir l'option "I will install the operating system later".

Etape 3 :

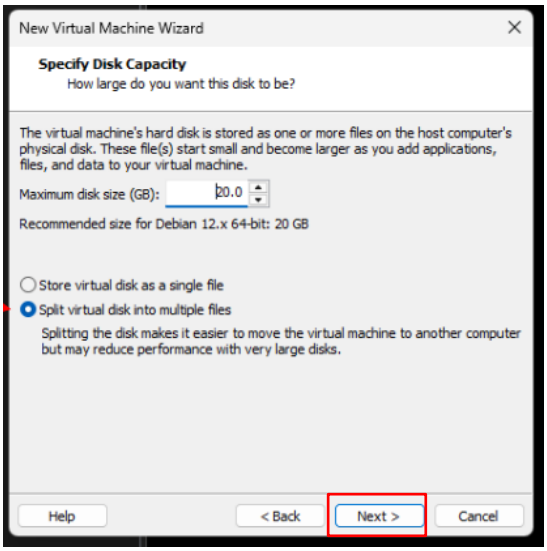


1. Choisir Linux et mettre la version « Debian 10. x64 ».

2. Changer le nom de la machine en mettant « SRV-WEB1 ».

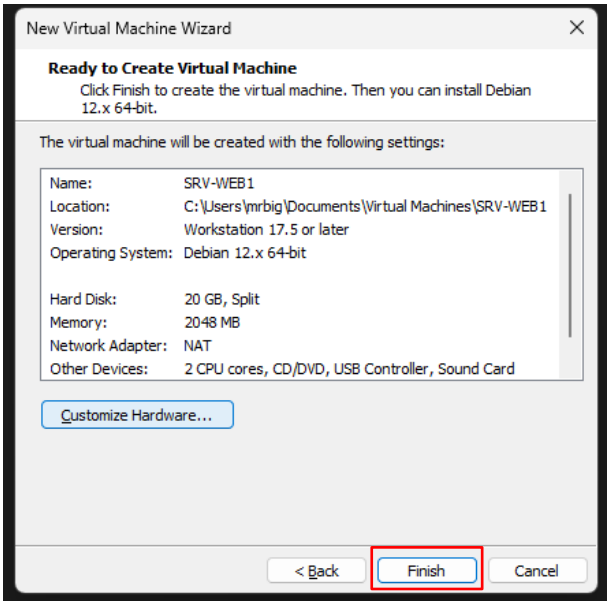


Etape 4 :

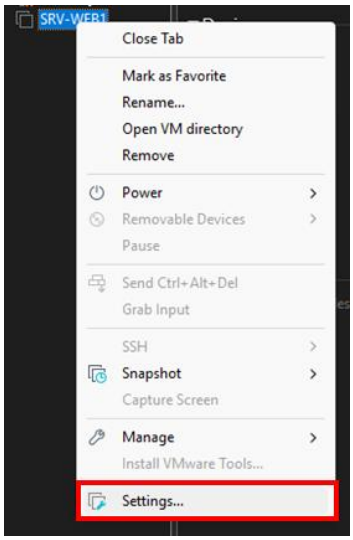


1. Le stockage de la machine virtuelle n'est pas à modifier tout comme les paramètres.

2. A la fin de cela, cliquer sur « Finish » afin de valider la création de la machine.

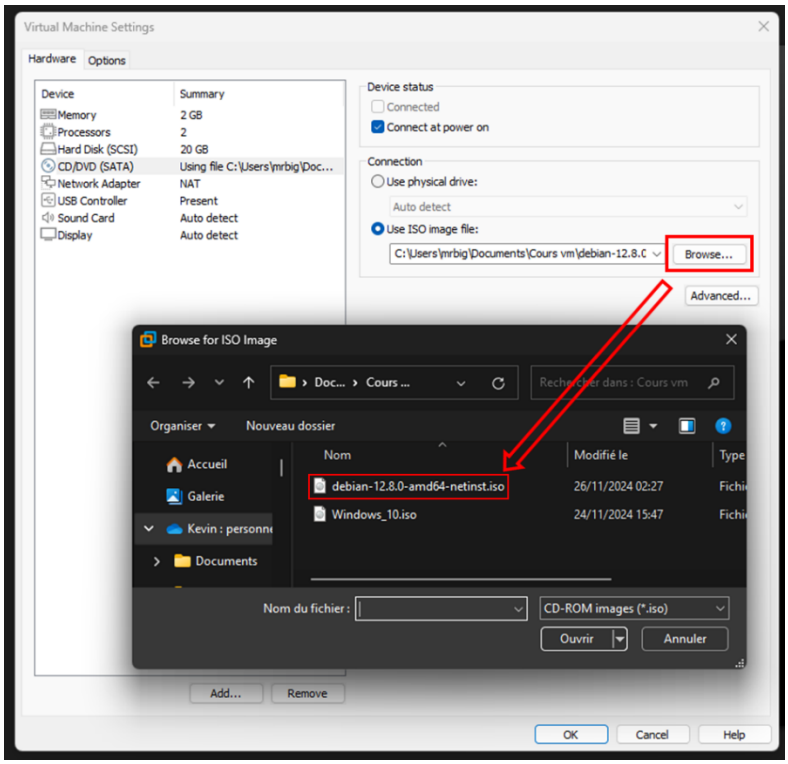


Etape 5 :

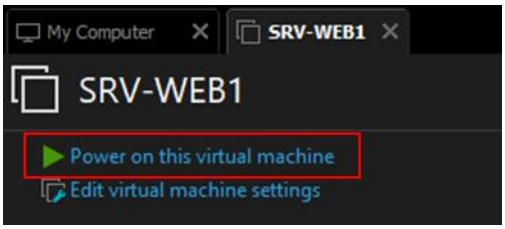


1. Allez dans les paramètres de la machine en faisant clique droit puis « settings ».

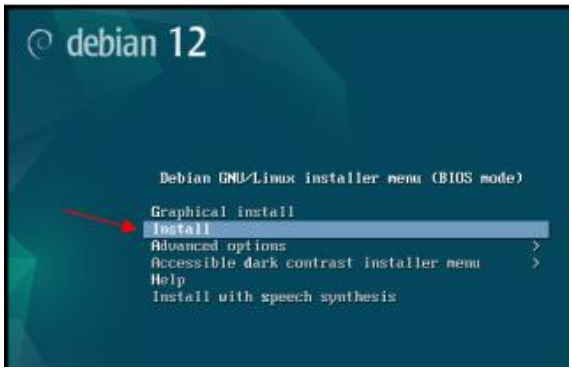
2. Rajouter le fichier iso dans « CD/DVD (SATA) ». Pour se faire, dans « Use ISO image file », choisir le fichier ISO du Debian souhaité. Puis on valide.



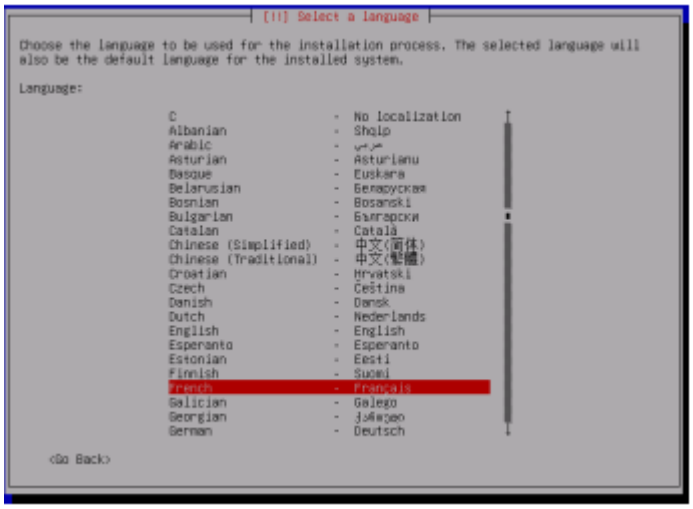
Etape 6 :



1. Allumer la machine.

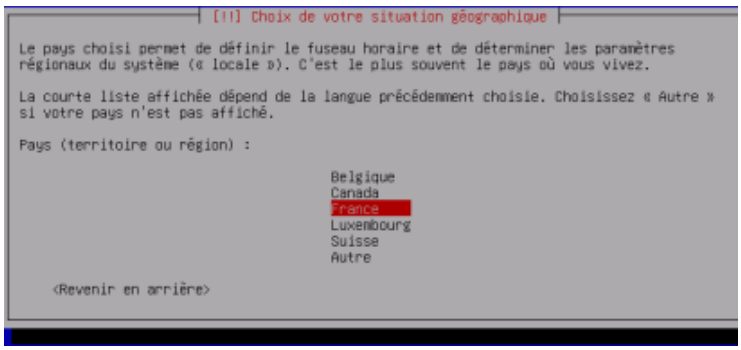


2. Choisir la méthode d'installation « Install ».



3. Choisir la langue en français.

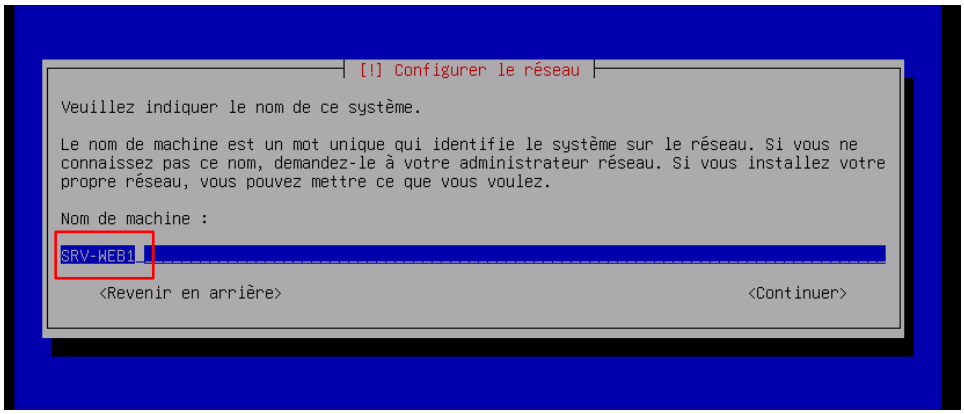
4. Mettre la situation géographique « France ».



Suite étape 6 :

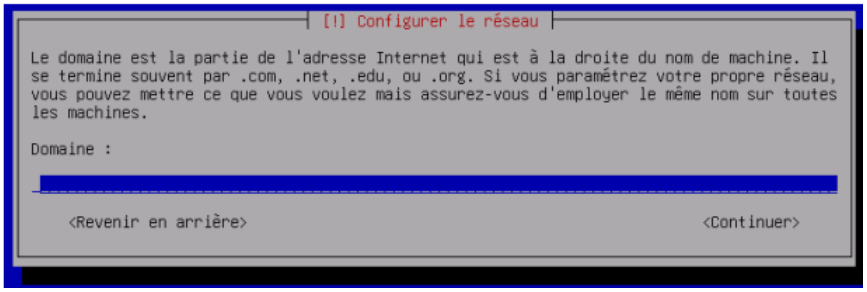
5. Mettre la disposition de clavier en français.

Etape 7 :

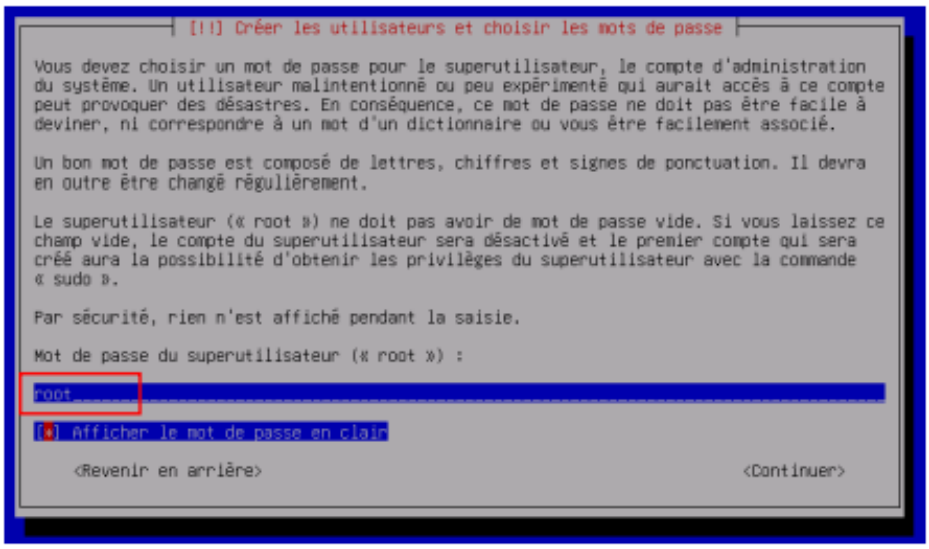


1. Rechanger le nom de la machine en « SRV-WEB1 ».

2. Ne mettre aucun domaine.



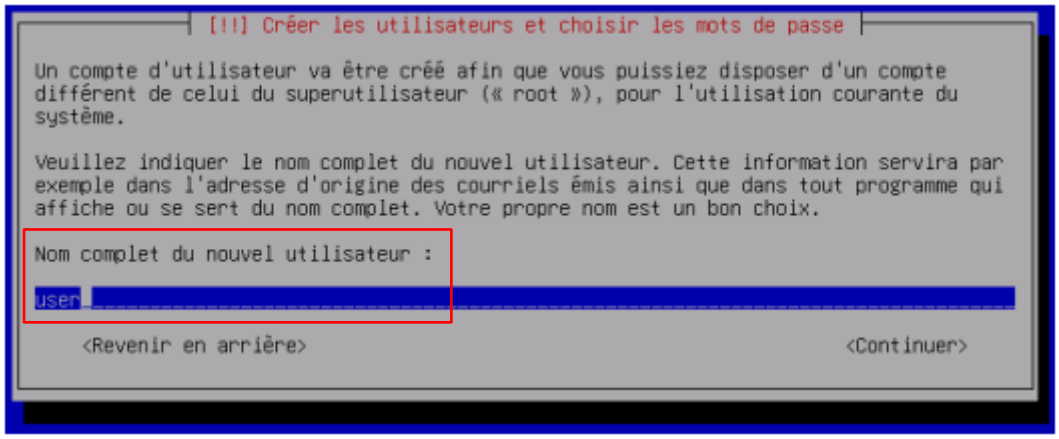
Suite étape 7 :



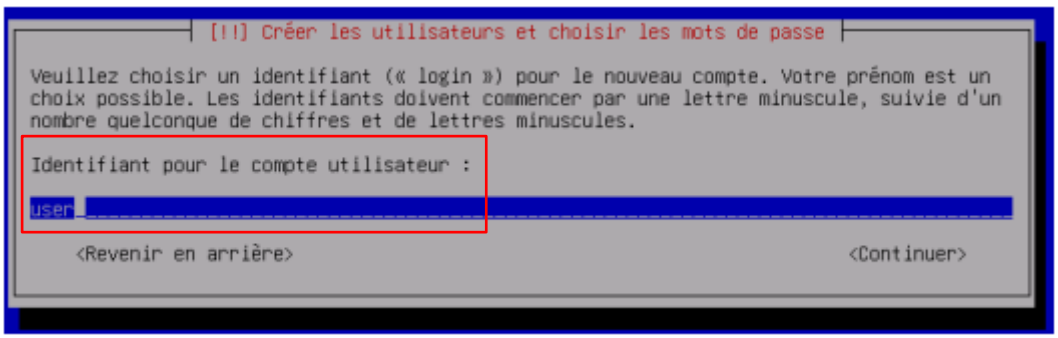
3. Pour le mot de passe superutilisateur mettre « root ».

Etape 8 :

1. Le nom du nouvel utilisateur en « user ».

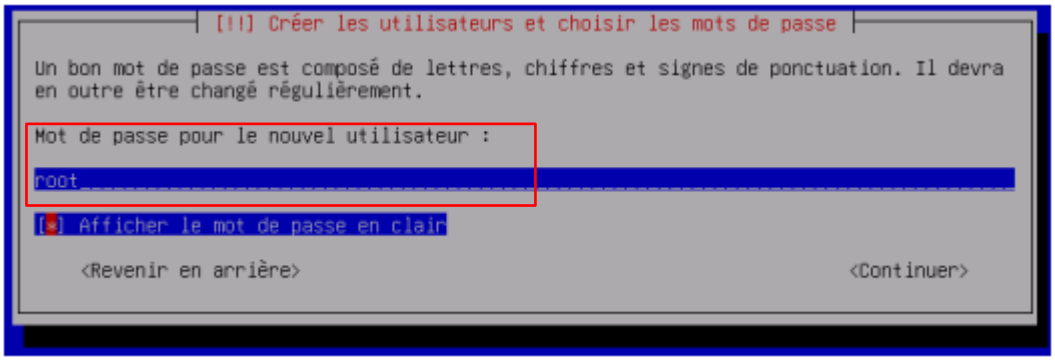


2. Le nom de l'identifiant en « user ».



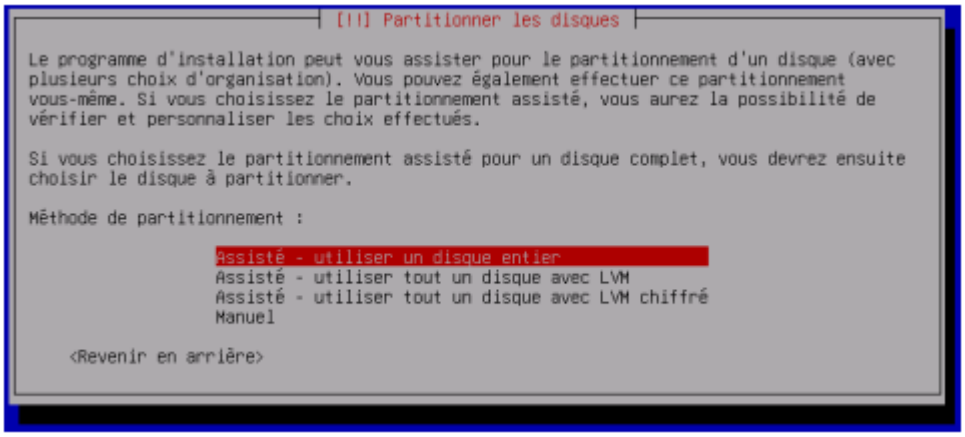
Suite étape 8 :

3. Ainsi que le mot de passe « root » pour l'utilisateur.

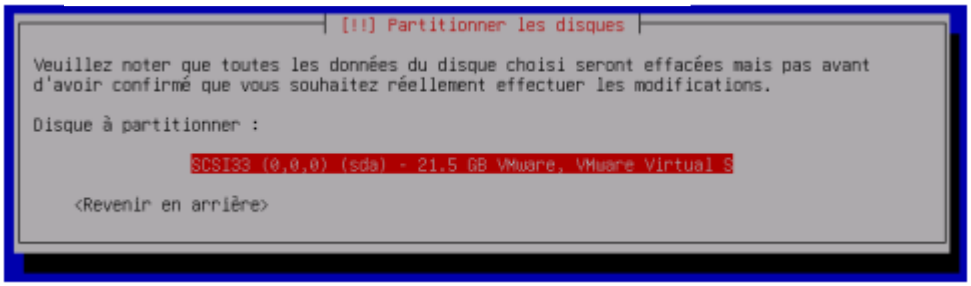


Etape 9 :

1. Pour la méthode de partitionnement mettre « Assisté, utiliser un disque entier »

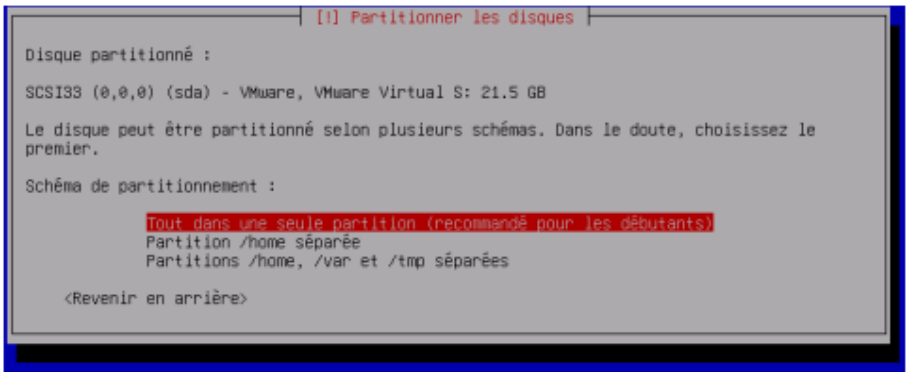


2. Choisir le disque (qu'un seul choix).

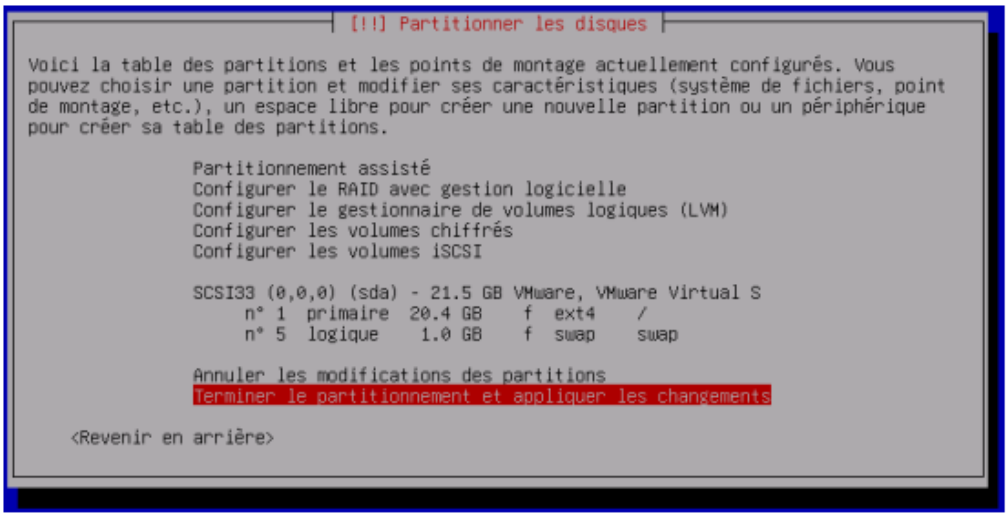


Suite étape 9 :

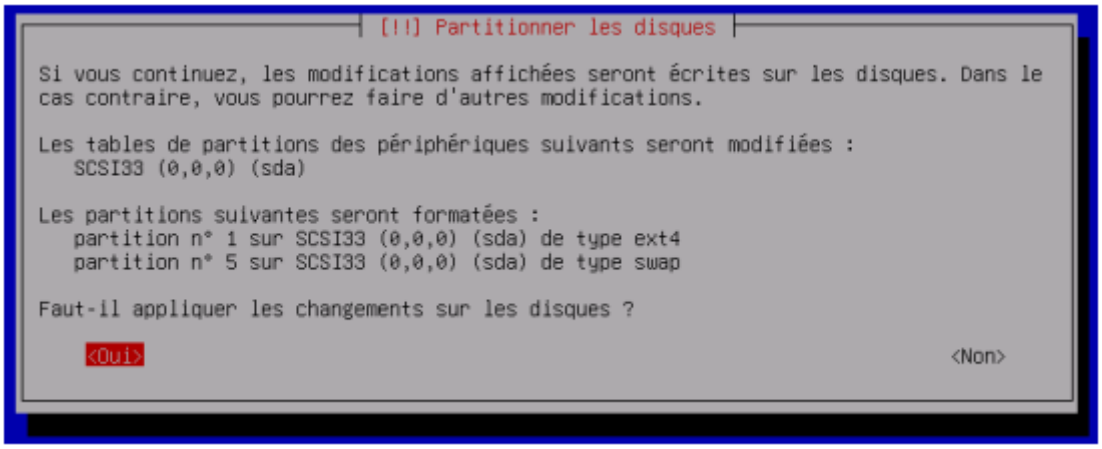
3. Pour le schéma de partitionnement mettre « Tout dans une seule partition ».



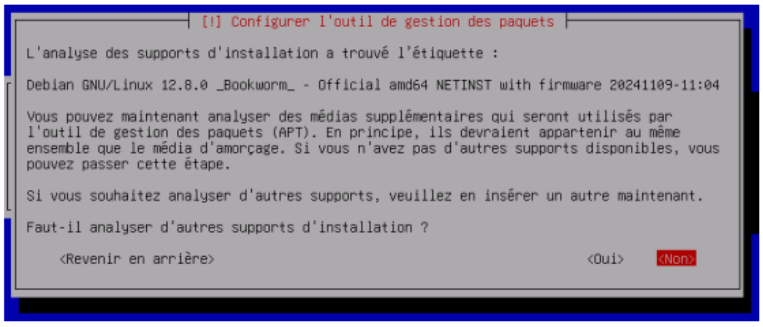
4. Terminer le partitionnement et appliquer les changements.



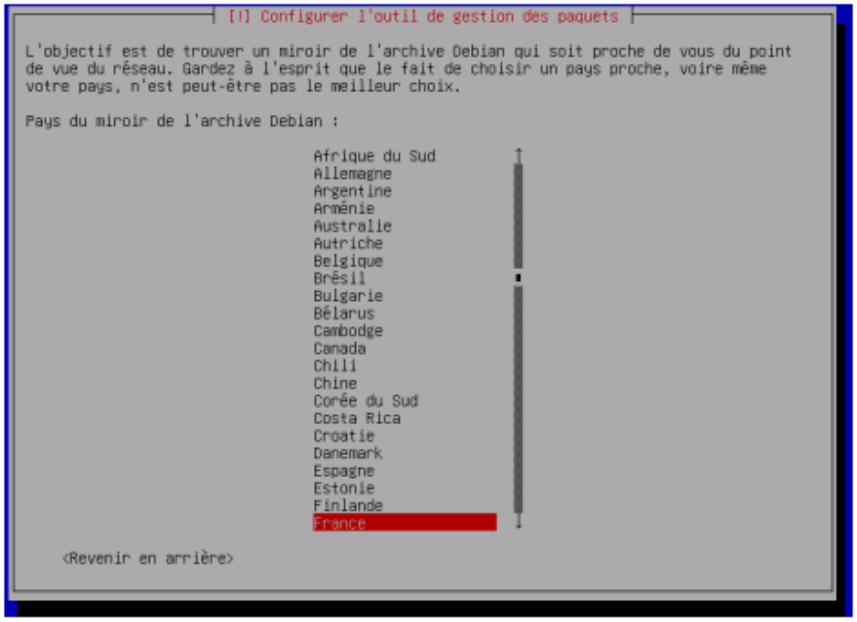
5. Ainsi, il faut accepter les changements sur le disque.



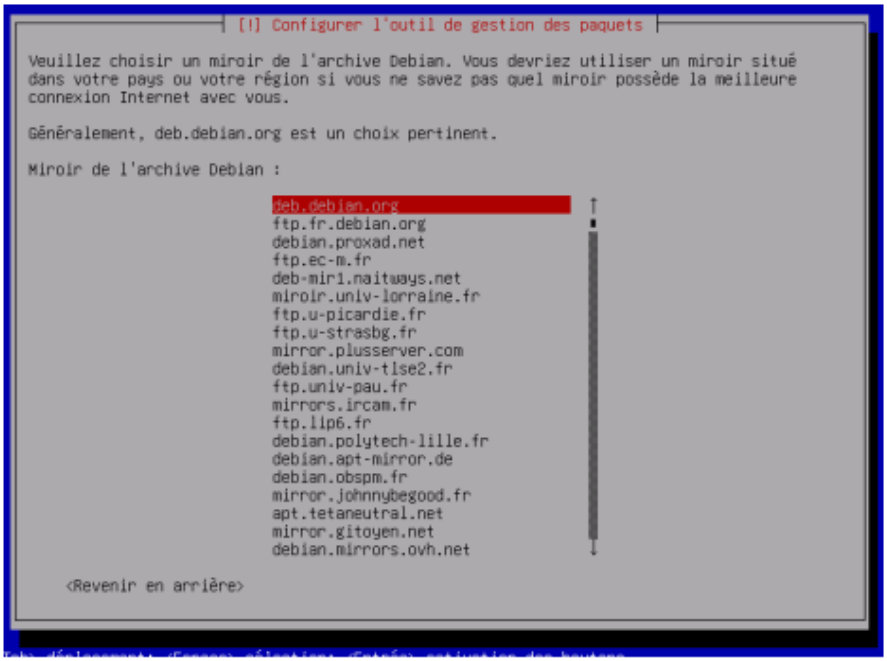
Etape 10 :



1. Refuser l'analyse des supports d'installation.



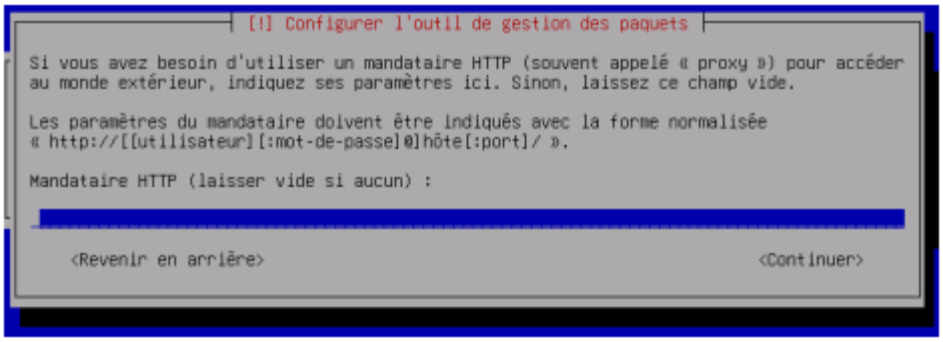
2. Choisir la France comme pays miroir de l'archive Debian.



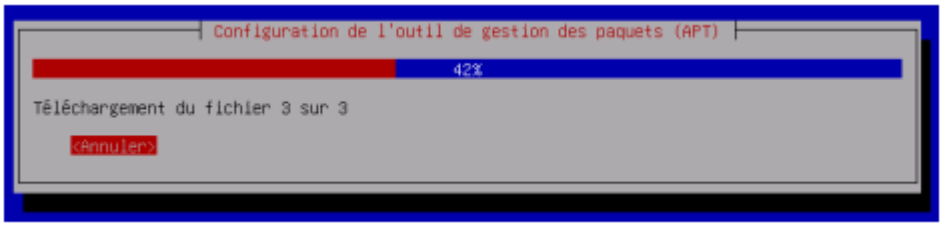
3. Enfin, choisir « deb.debian.org » comme miroir de l'archive de Debian.

Etape 11 :

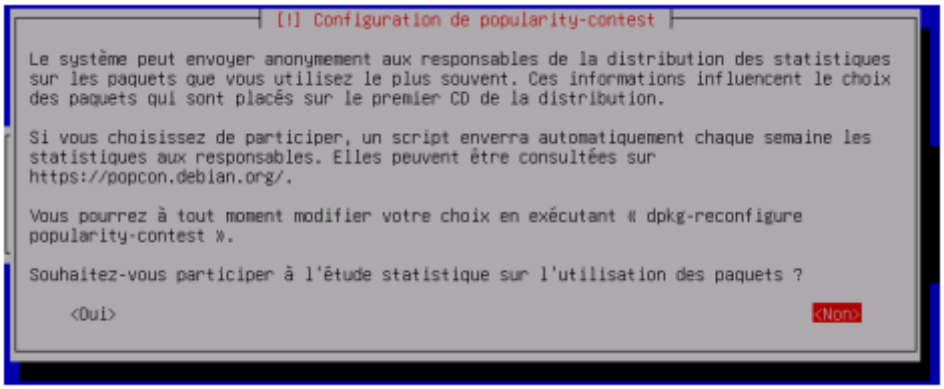
1. Ne mettre aucun mandataire HTTP.



2. On laisse les paquets se télécharger.

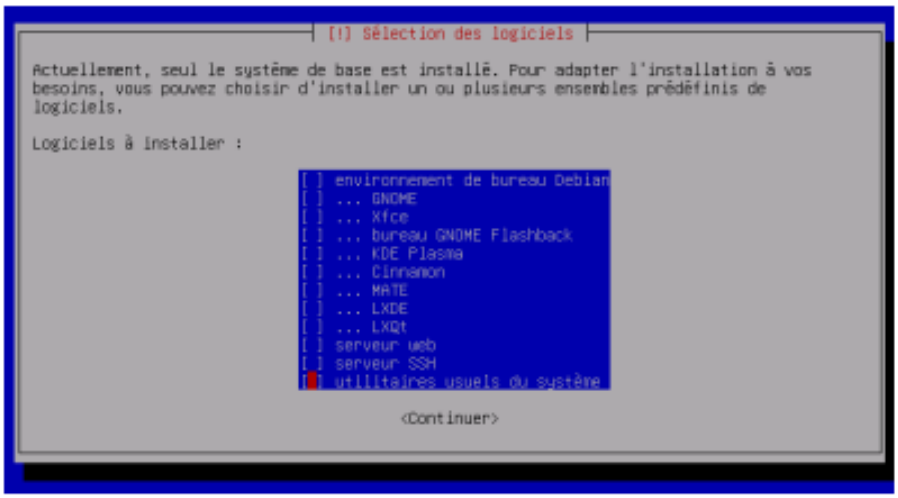


3. Mettre « non » pour la participation à l'étude des statistiques sur l'utilisation des paquets.

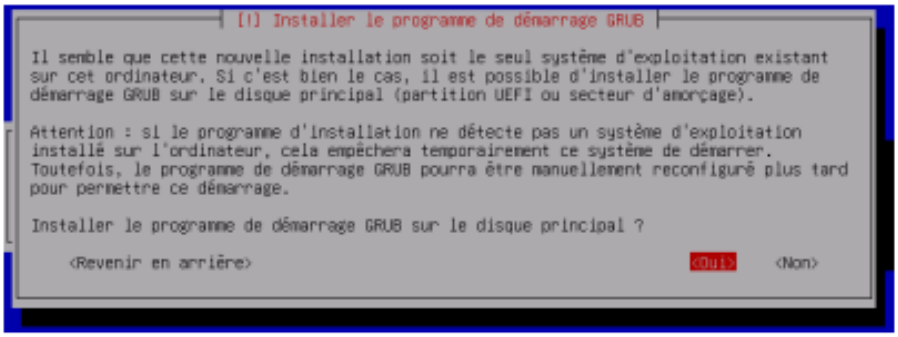


Etape 12 :

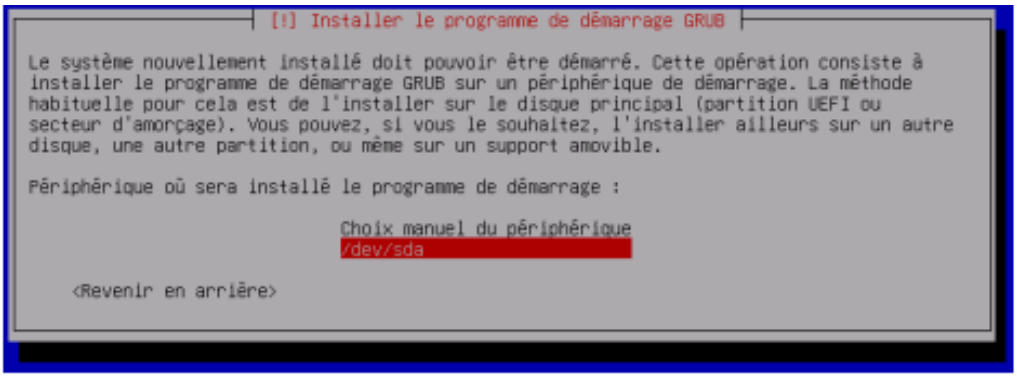
1. Décocher les logiciels à installer par défaut « environnement de bureau Debian, GNOME, utilitaires usuels du système ».



2. Installer le programme de démarrage GRUB sur le disque



3. Choisir le périphérique « /dev/sda/ ».



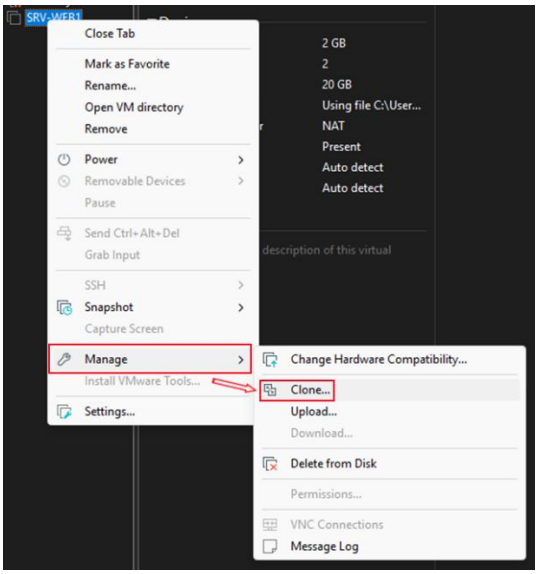
Etape 13 :

1. On redémarre la machine virtuelle Debian

2. L'installation de la machine virtuelle Debian sans interface graphique SRV-WEB1 est terminée.

b. *Duplication de la machine SRV-WEB1 pour créer la machine HAProxy1*

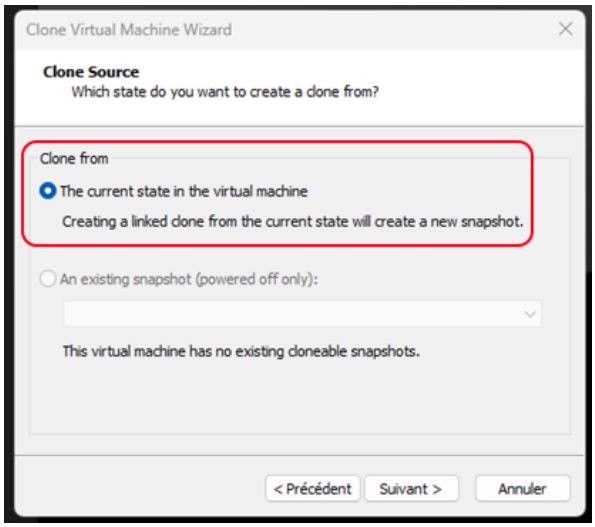
Etape 1 :



Faire un clique droit sur la machine SRV-WEB1, ensuite aller sur manage puis cliquer sur « Clone... ».

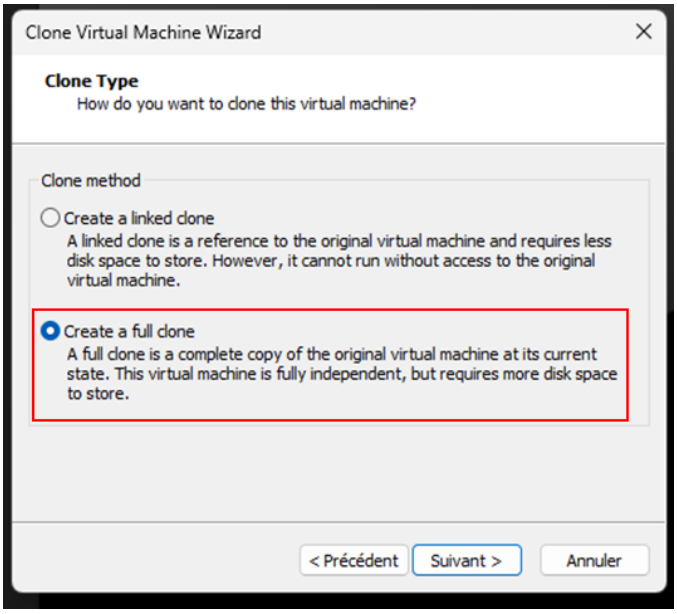
Etape 2 :

1. Cliquer sur suivant



2. Choisir « The current state in the virtual machine » puis faire suivant.

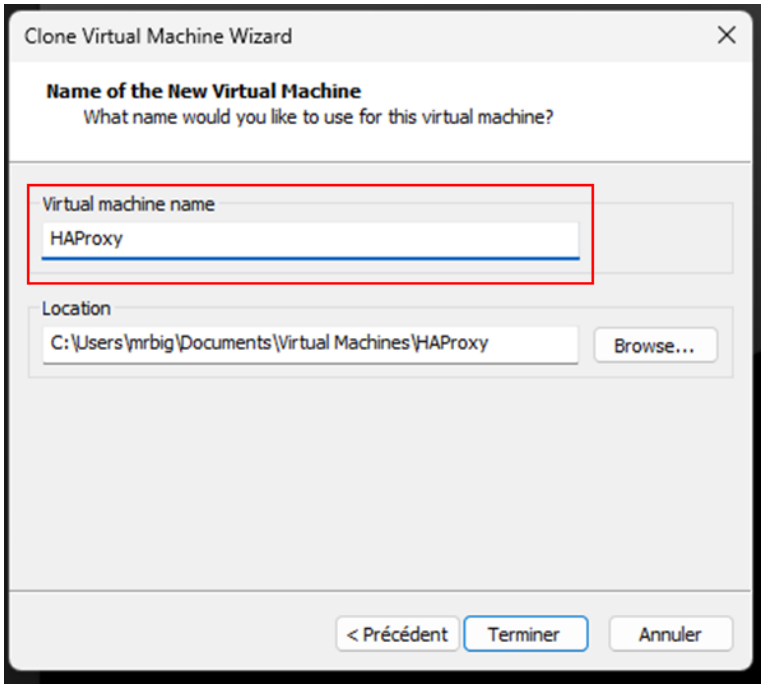
Etape 3 :



Choisir « Create a full clone » et cliquer sur suivant.

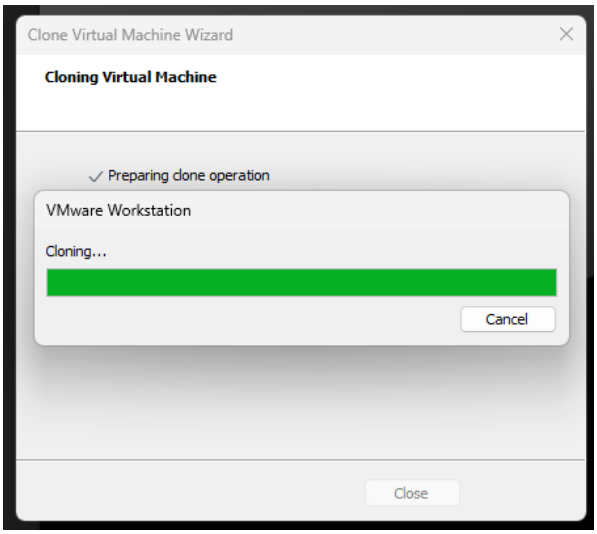
Etape 4 :

Mettre le nom de la machine HAProxy et ensuite faire terminer.

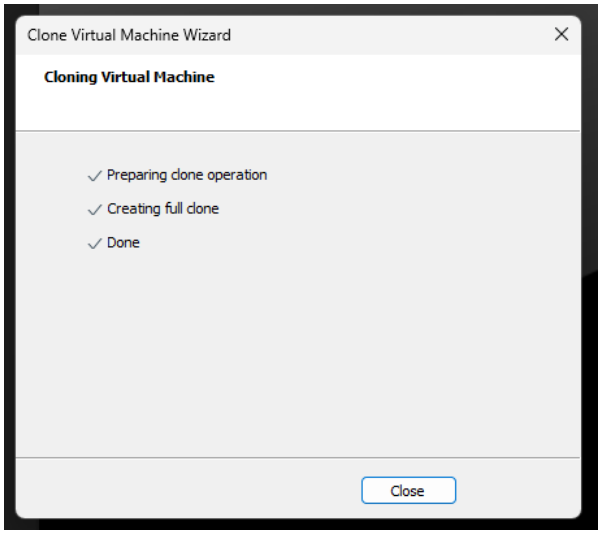


Etape 5 :

1. Le clonage est en cours de création



2. Le clonage est terminé.

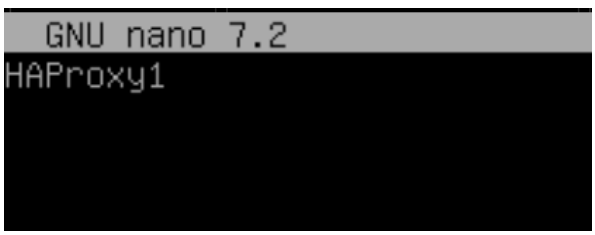


Etape 6 :

1. Allumez la machine HAProxy1, taper la commande suivante pour pouvoir la renommer.

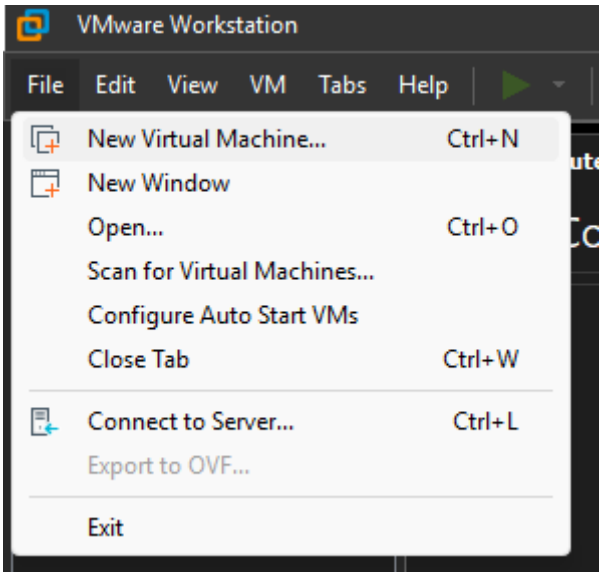
nano /etc/hostname

2. Ecrire HAProxy1 et valider la modification.



c. *Installation de la machine virtuelle Client Windows + configuration de l'adresse IP*

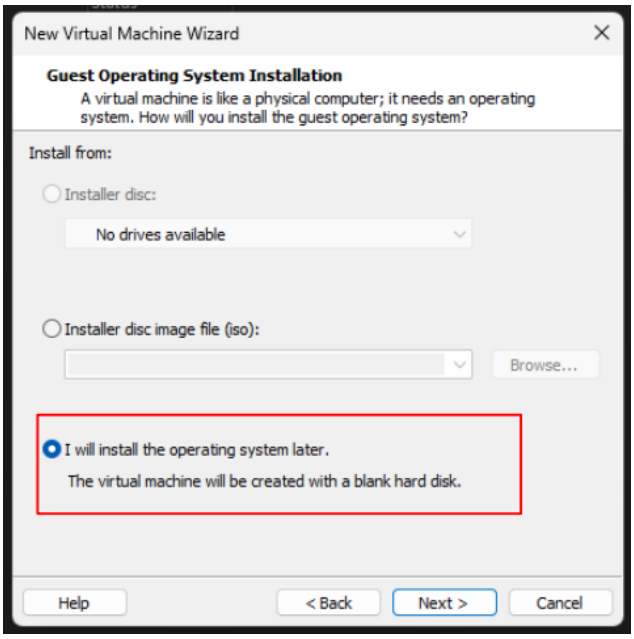
Etape 1 :



1. Appuyer sur file puis sur « New Virtual Machine »

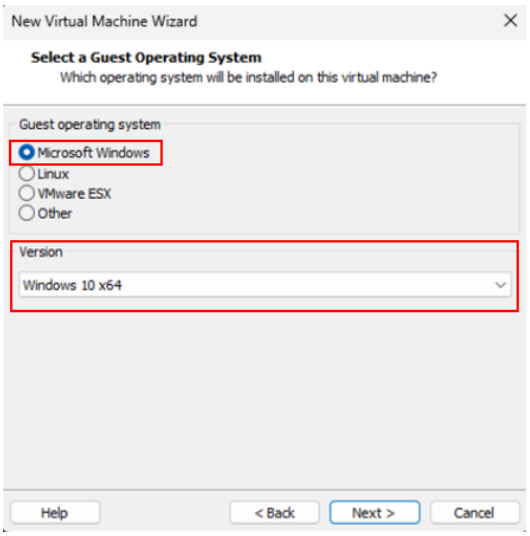
2. Choisir le type de la machine « Typical (recommanded) ».

Etape 2 :

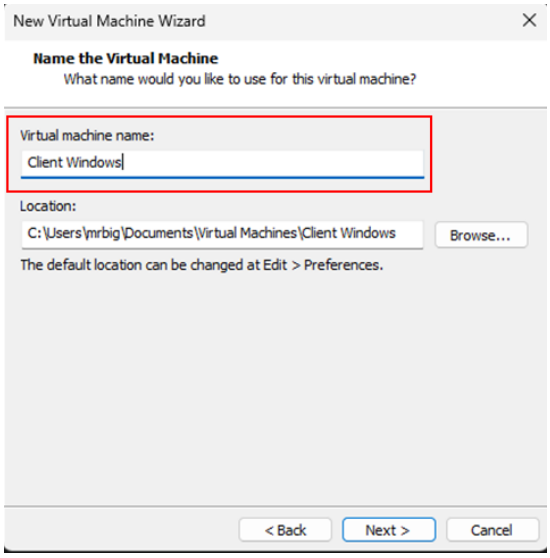


Cocher le dernier point « I will install the operating system later ».

Etape 3 :

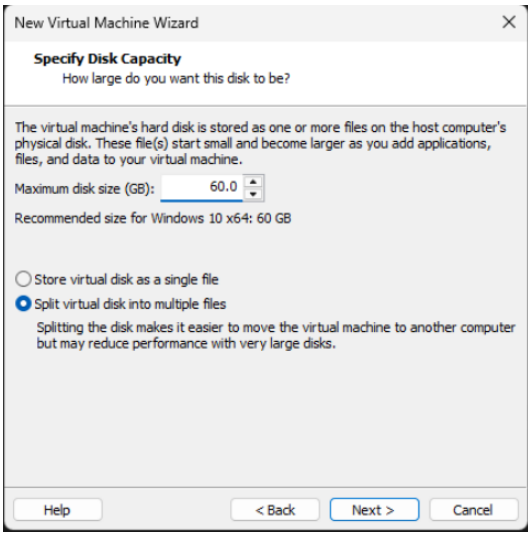


1. Choisir Microsoft Windows et mettre la version « Windows 10.



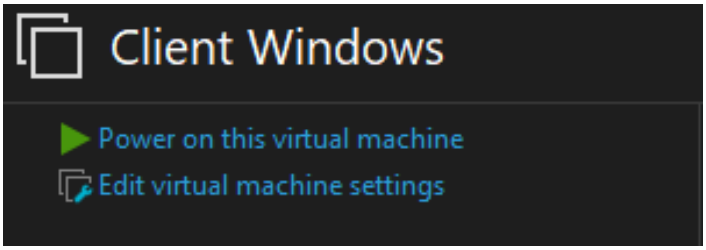
2. Changer le nom de la machine en mettant « Client Windows ».

Etape 4 :



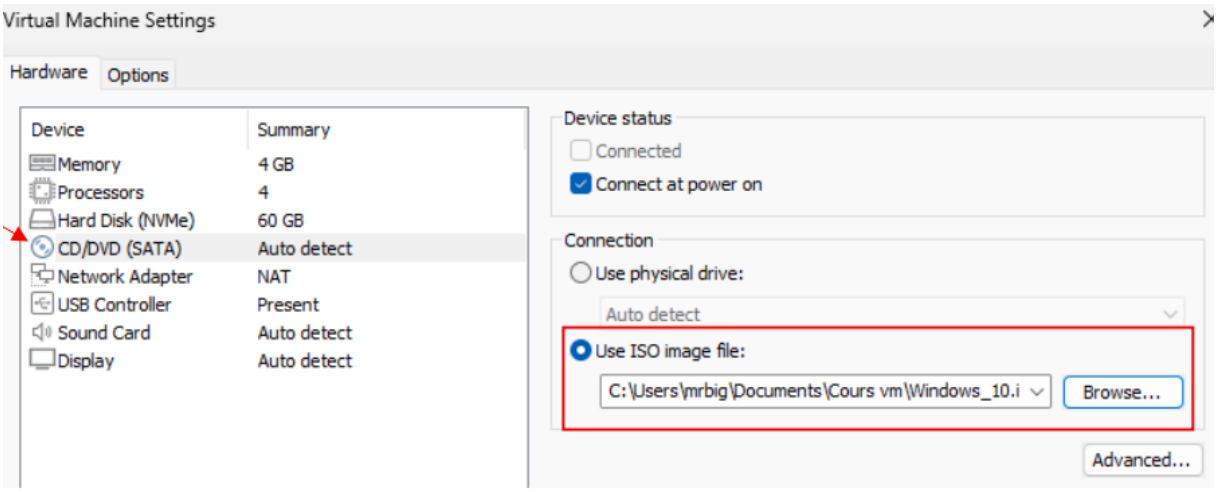
1. Le stockage de la machine virtuelle n'est pas à modifier et ne pas changer les paramètres personnalisés.

2. A la fin de cela, il sera possible d'allumer la machine.



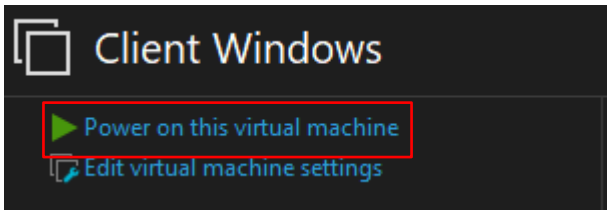
Etape 5 :

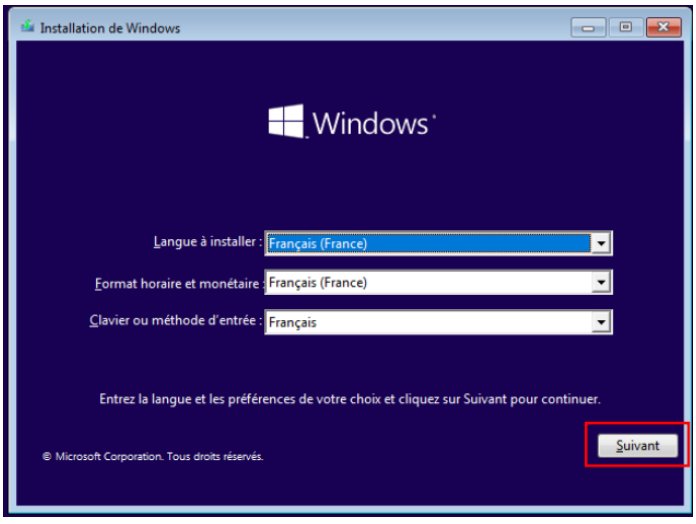
Allez dans les paramètres de la machine « settings » et rajouter le fichier iso dans « CD/DVD (SATA) ». Ensuite dans « Use ISO image file », choisir le fichier ISO du Windows souhaité.



Etape 6 :

1. Allumer la machine.

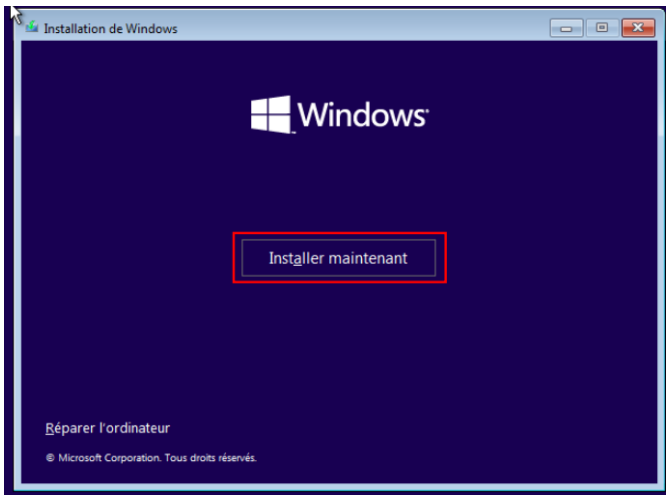


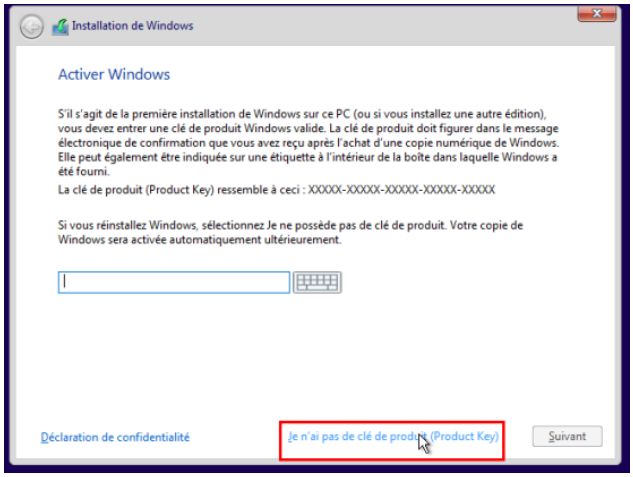


2. Choisir la langue, le format horaire et monétaire et clavier ou méthode d'entrée en « Français ». Appliquer et ensuite cliquer sur « Suivant ».

Etape 7 :

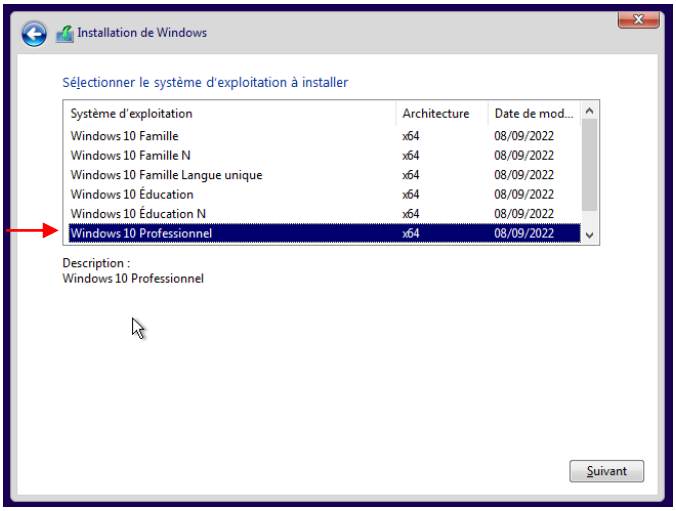
1. On clique sur « Installer maintenant ».





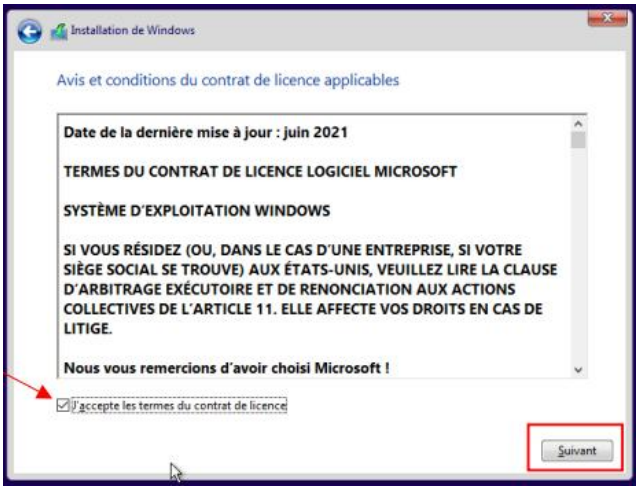
2. Pour activer Windows, choisir « Je n'ai pas de clé de produit (Product Key) ».

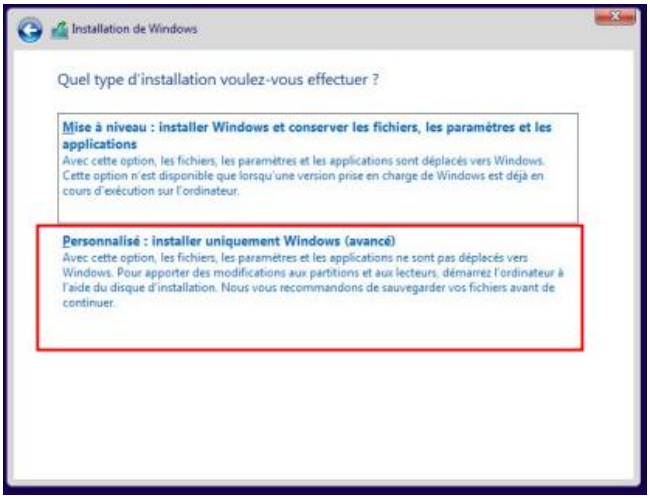
3. Ensuite choisir le système d'exploitation à installer « Windows 10 Professionnel ».



Etape 7 (Suite) :

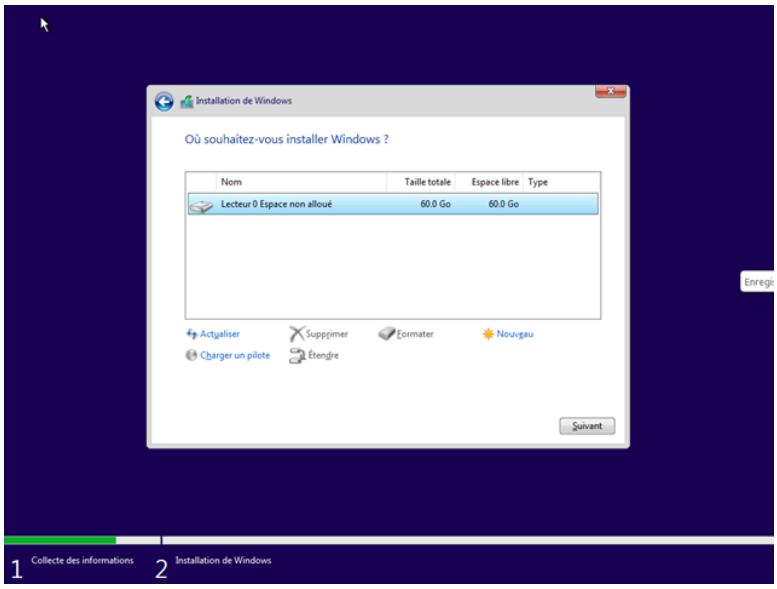
1. Accepter les termes du contrat de licence.



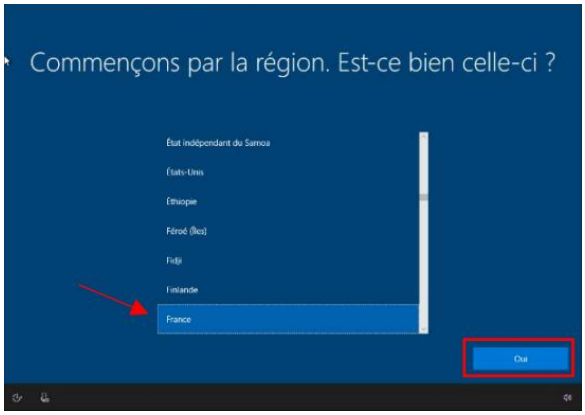


2. Choisir le type d'installation « Personnalisé : installer uniquement Windows (avancé) ».

3. Choisir le lecteur où installer Windows et ensuite l'installation de Windows commencera.

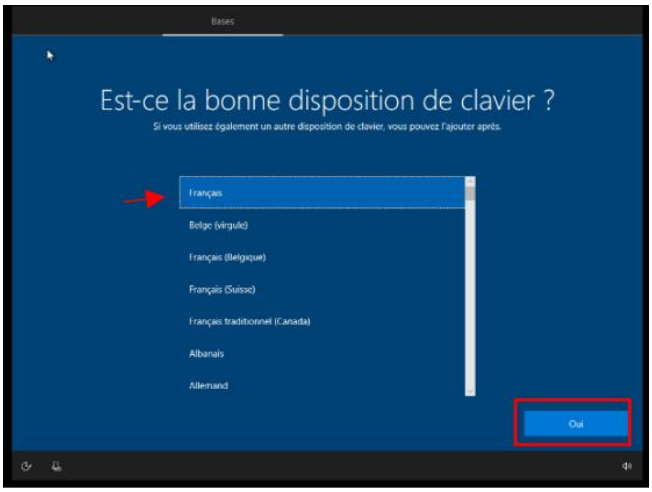


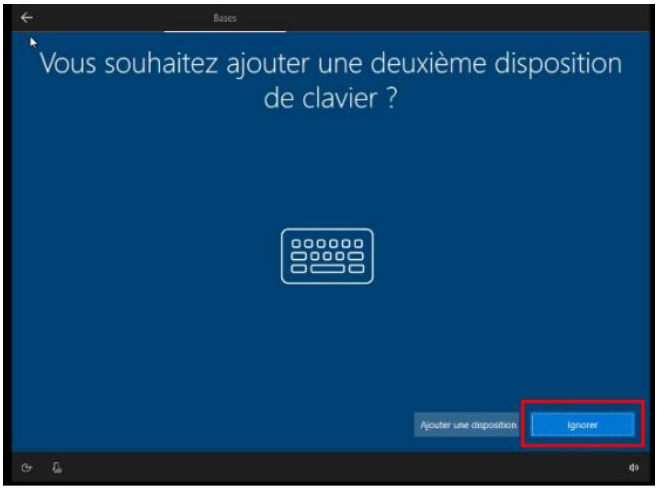
Etape 8 :



1. La machine virtuelle Windows redémarre et ensuite il faut mettre la situation géographique « France ».

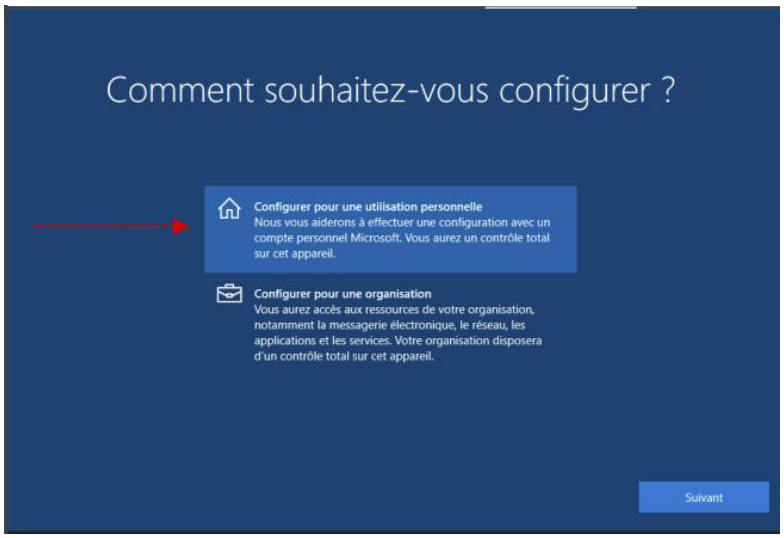
2. Mettre la bonne disposition de clavier en « Français ».



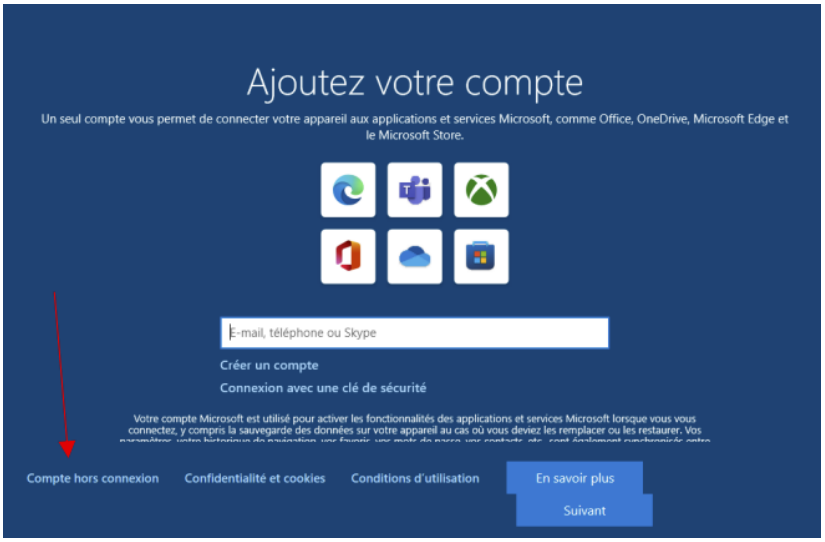


3. Enfin, ignorer l'ajout d'une deuxième disposition de clavier.

Etape 9 :

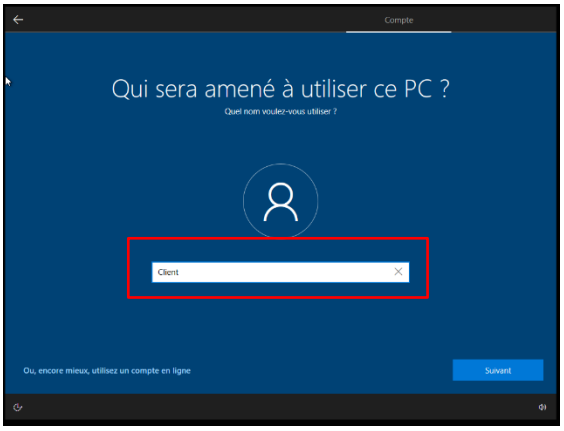


1. Choisir la configuration pour une utilisation personnelle.



2. Ensuite choisir compte hors connexion.

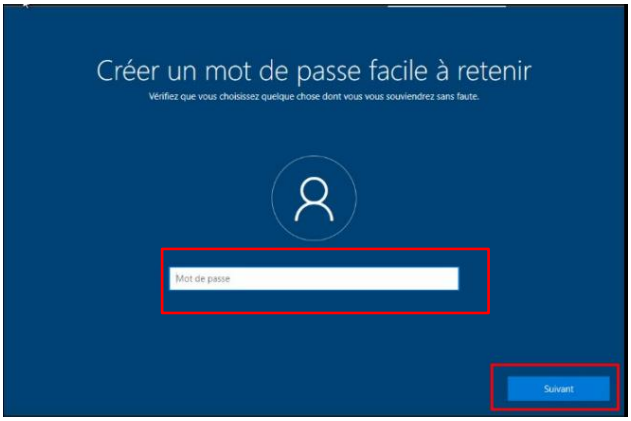
Etape 10 :



1. Mettre le nom du nouvel utilisateur « Client ».

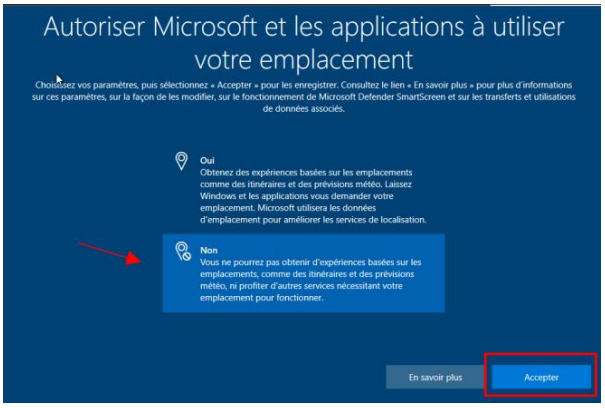
Suite étape 10 :

2. Ne pas mettre de mot de passe et confirmer avec « suivant ».

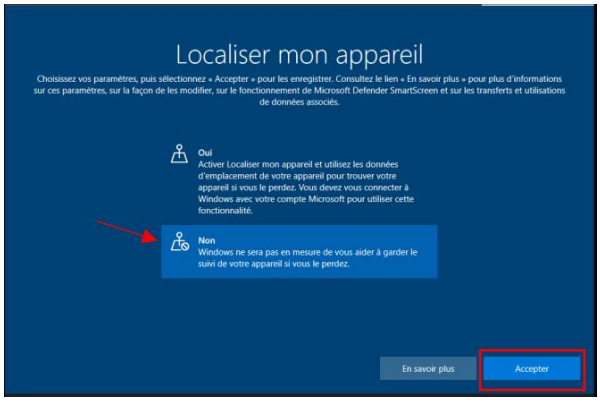


Etape 11 :

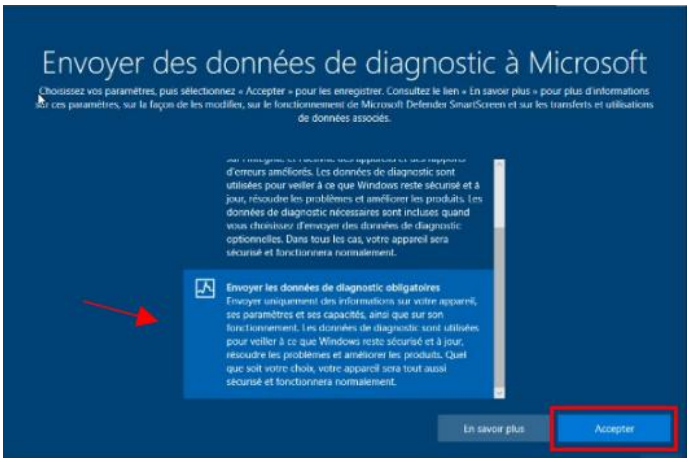
1. Refuser que Microsoft et les applications utilisent l'emplacement de l'ordinateur et le localise



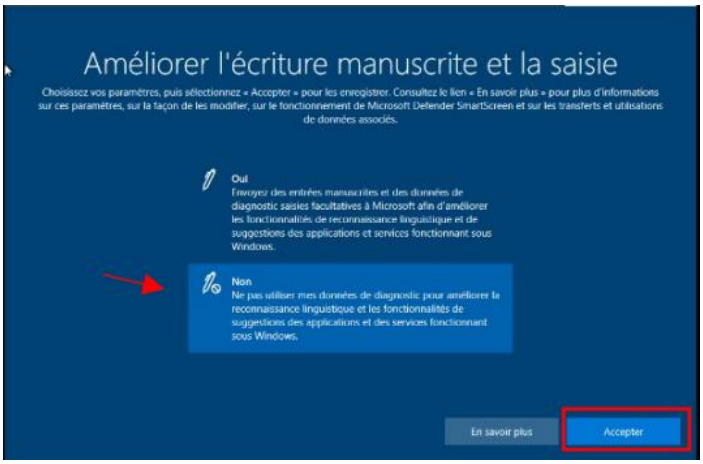
2. Choisir qu'il n'y a que les données de diagnostic obligatoire qui soit envoyées.



3. Choisir qu'il n'y a que les données de diagnostic obligatoire qui soit envoyées.

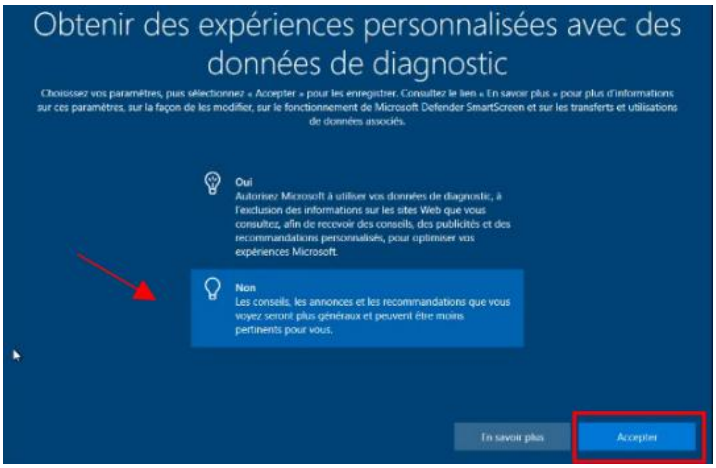


4. Ne pas aider à l'amélioration de l'écriture manuscrite et à la saisie.

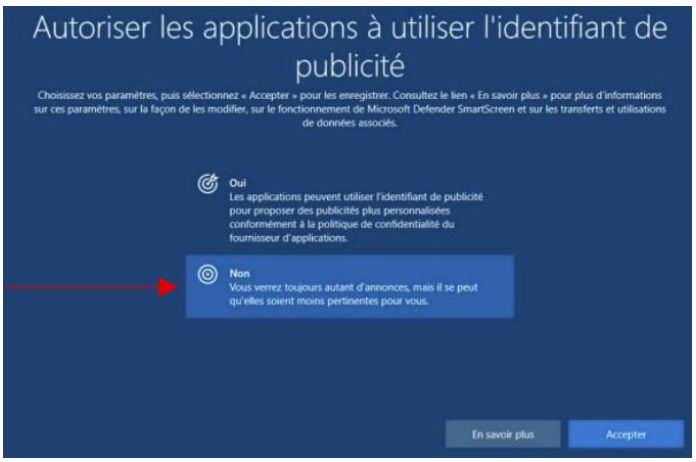


Suite étape 11 :

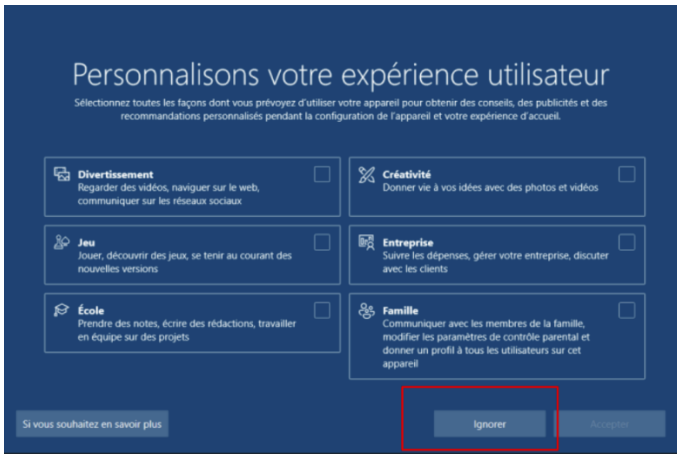
5. Enfin ne pas obtenir des expériences personnalisées avec des données de diagnostic.



6. Refuser l'autorisation aux applications à utiliser l'identifiant de publicité.

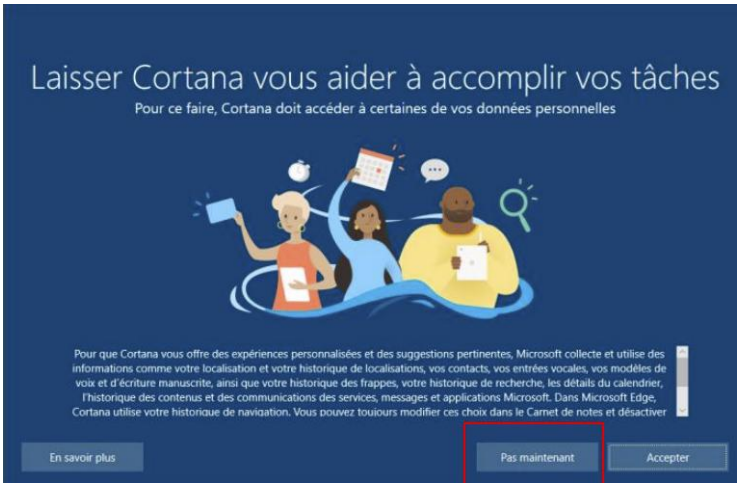


Etape 12 :



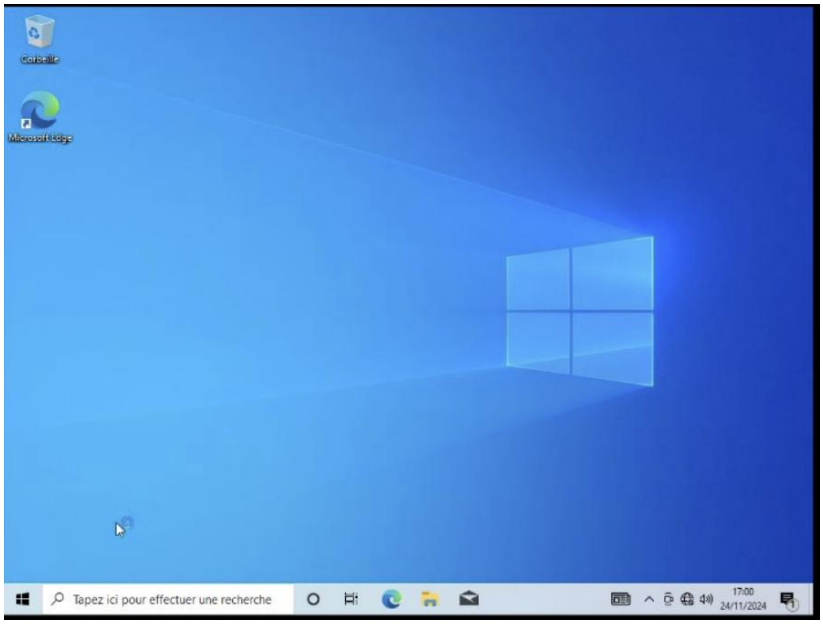
1. Ignorer la personnalisation de l'expérience utilisateur.

2. Ne pas laisser Cortana accomplir les tâches en choisissant « Pas maintenant ».

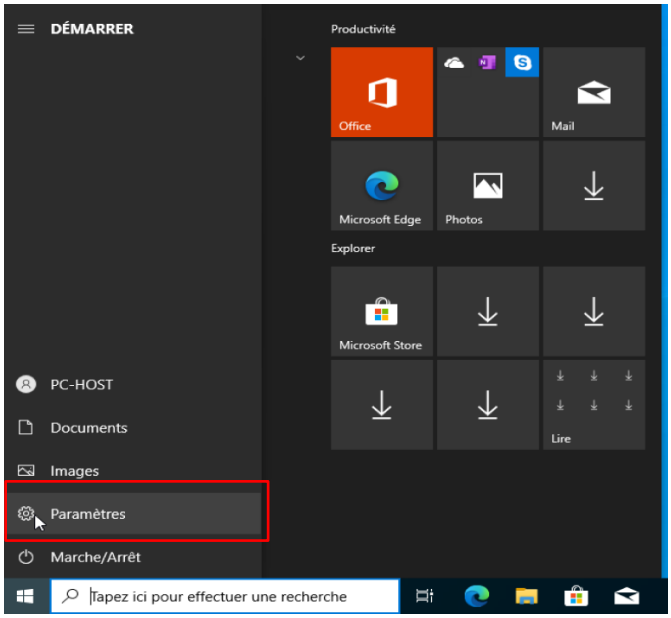


Etape 13 :

L'installation de la machine virtuelle Client Windows est terminée.

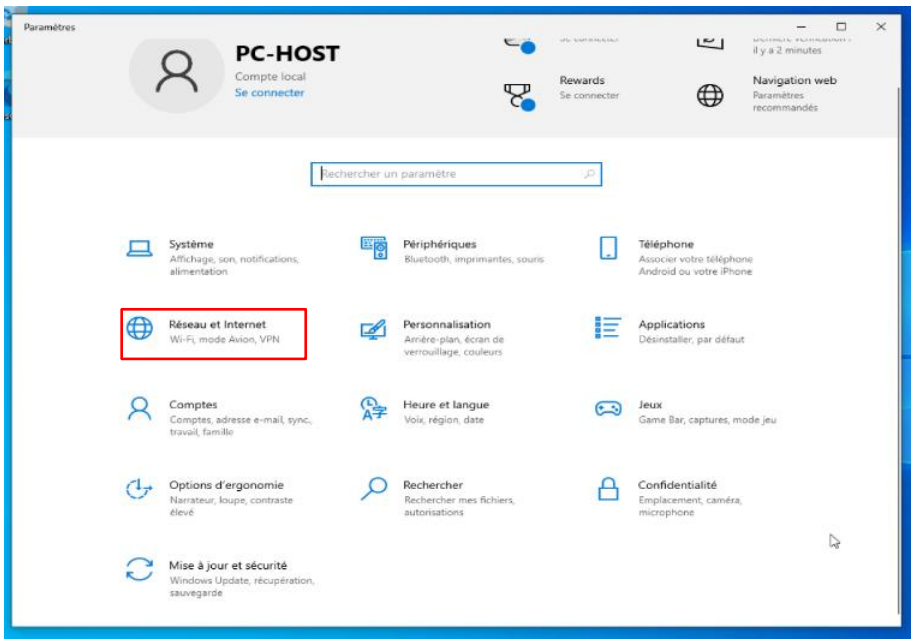


Etape 14 :



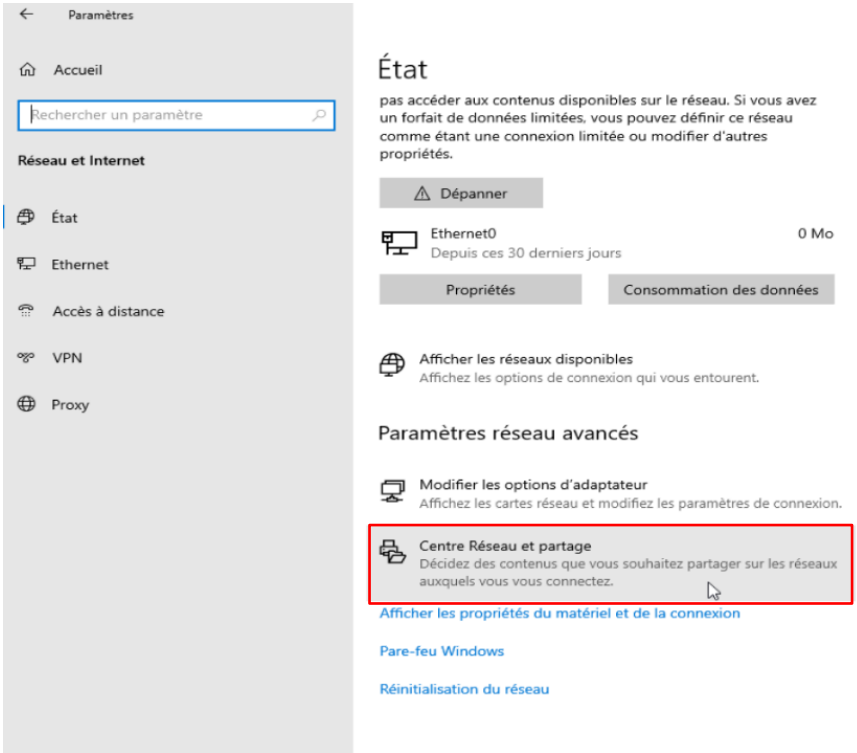
1. Concernant la configuration de l'adresse IP, il faut aller dans les paramètres en cliquant sur la touche Windows et cliquer sur l'engrenage.

Etape 15 :

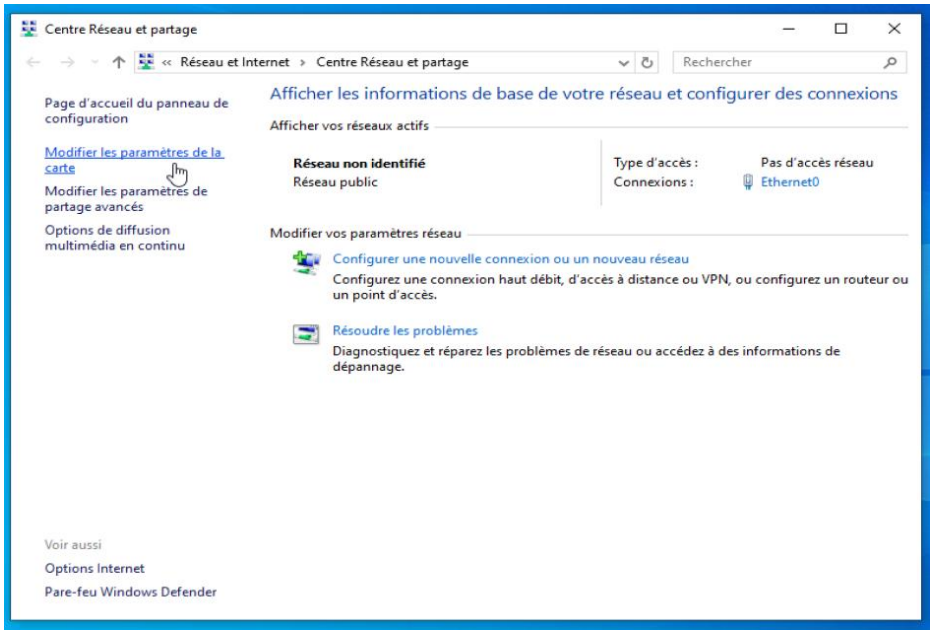


1. A la suite de cela, il faut se rendre dans Réseau et Internet.

2. Cliquer sur Centre Réseau et Partage.

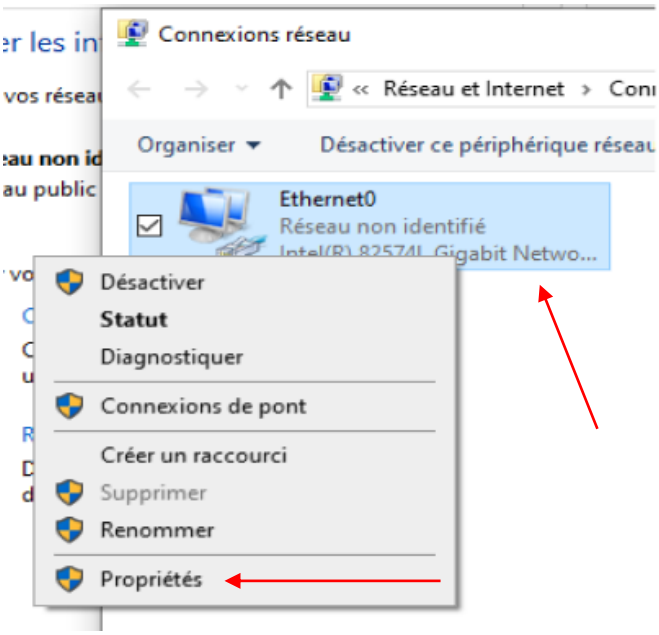


Etape 16 :

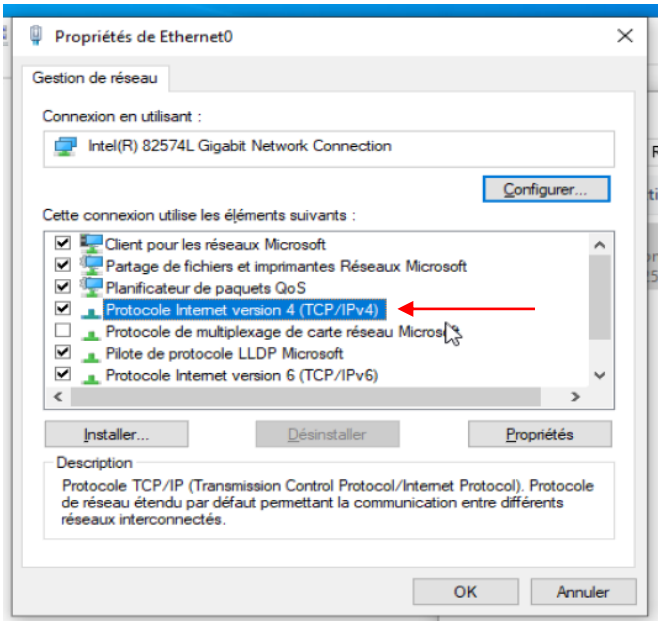


1. Cliquer sur modifier les paramètres de la carte.

2. Clique droit, puis propriété sur notre carte réseau.



Etape 17 :

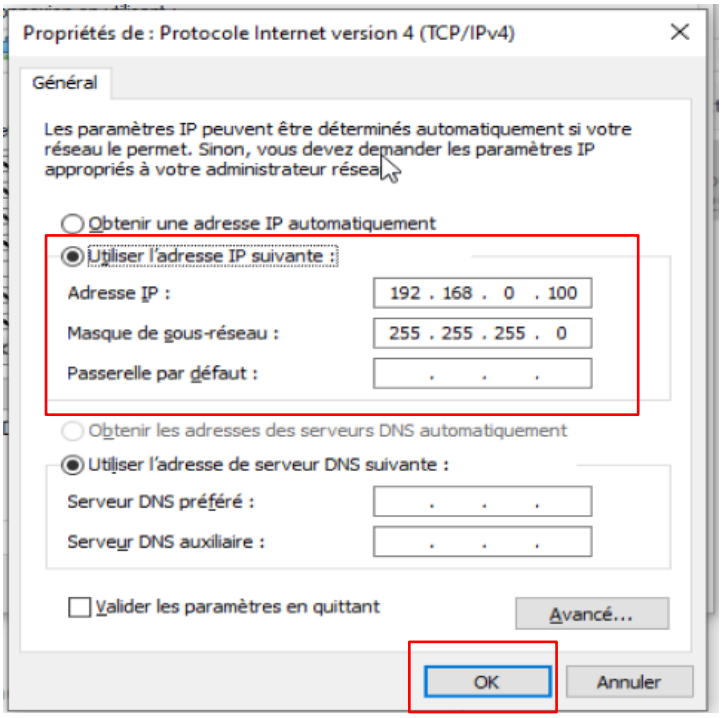


1. Double clique sur le « Protocol Internet version 4 (TCP/IPv4) ».

2. Cliquer sur « Utiliser l'adresse IP suivante » et rentrer celle-ci :

Adresse IP : 192.168.0.100
Masque de sous-réseau : 255.255.255.0

Cliquer sur « OK » et cela sauvegardera les modifications apportées.

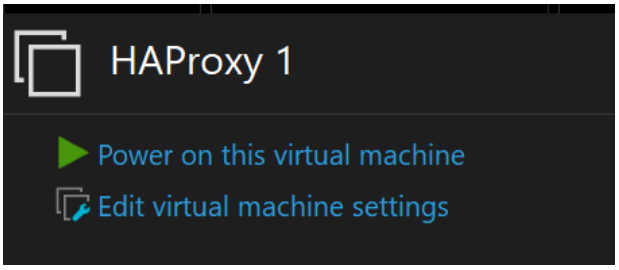


L'installation de la machine Client Windows et la configuration de son adresse IP sont terminées.

2. *Mise en place d'HAProxy pour la machine HAProxy DEBIAN*

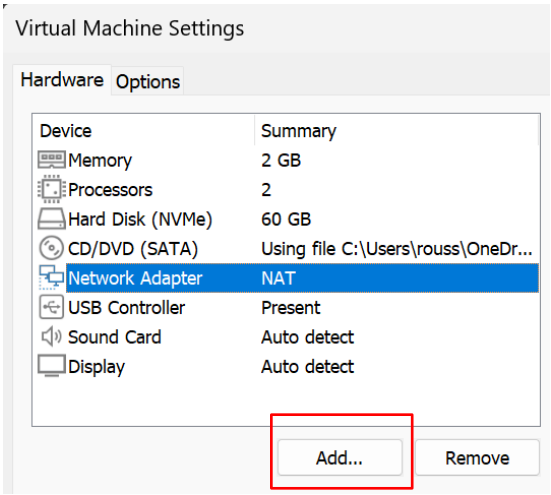
a. *Configuration des adresses IP HAProxy 1*

Etape 1 :

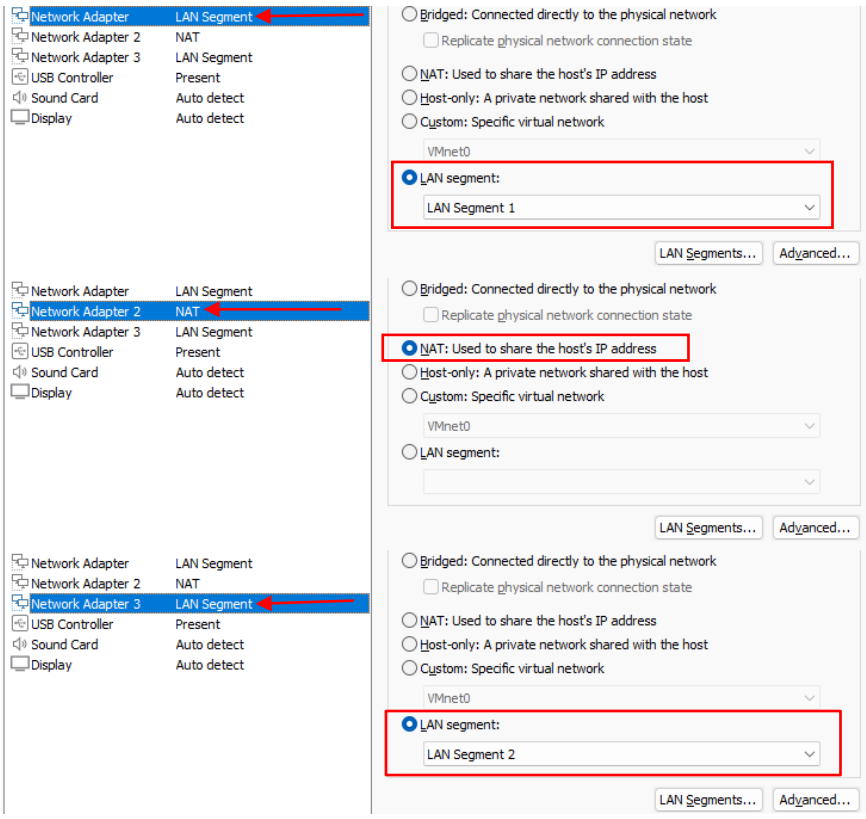


1. Il faut se rendre dans les paramètres de la machine virtuelle HAProxy1, en cliquant sur « settings ».

2. Ensuite rajouter 2 « Network adapter » en cliquant sur « Add... ».



Suite étape 1 :



3. Mettre le premier en LAN segment.
« Lan segment 1 »

Le second en NAT.

Le troisième en LAN segment

« Lan segment 2 »

Etape 2 :

1. Allumer la machine, taper la commande nano /etc/network/interfaces.

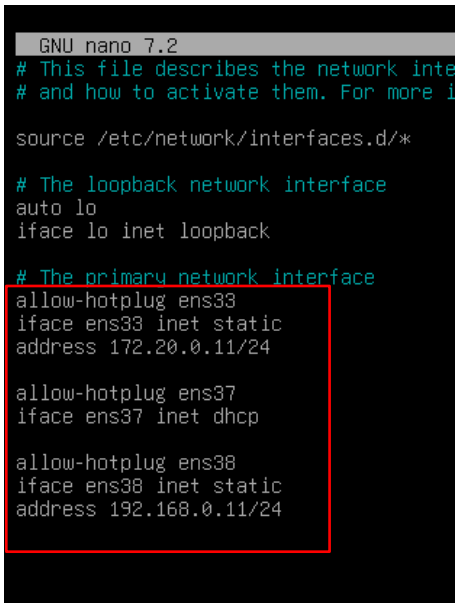
```
root@HAProxy1:~# nano /etc/network/interfaces
```

2. Modifier les lignes suivantes :

Allow-hotplug ens33
Iface ens33 inet static
Address
172.20.0.11/24

Allow hotplug ens37
Iface ens37 inet dhcp

Allow hotplug ens38
Iface ens38 inet static
Address
192.168.0.11/24



Etape 3 :

```
root@HAProxy1:~# ifdown ens33
root@HAProxy1:~# ifdown ens37
Killed old client process
Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens37/00:0c:29:6e:ea:d6
Sending on   LPF/ens37/00:0c:29:6e:ea:d6
Sending on   Socket/fallback
DHCPRELEASE of 192.168.42.137 on ens37 to 192.168.42.254 port 67
root@HAProxy1:~# ifdown ens38
```

1. Faire ifdown sur ens33 ens37 et ens38

2. Puis ifup sur ens33 ens37 et ens38.

```
root@HAProxy1:~# ifup ens33
root@HAProxy1:~# ifup ens37
Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens37/00:0c:29:6e:ea:d6
Sending on   LPF/ens37/00:0c:29:6e:ea:d6
Sending on   Socket/fallback
DHCPDISCOVER on ens37 to 255.255.255.255 port 67 interval 6
DHCPOFFER of 192.168.42.137 from 192.168.42.254
DHCPREQUEST for 192.168.42.137 on ens37 to 255.255.255.255 port 67
DHCPACK of 192.168.42.137 from 192.168.42.254
bound to 192.168.42.137 -- renewal in 879 seconds.
root@HAProxy1:~# ifup ens38
```

Etape 4 :

Puis vérifier à l'aide de la commande ip a qu'il y a bien une adresse IP inet Static 172.20.0.11/24 pour ens33, une adresse IP inet DHCP pour ens37 et une adresse IP inet Static 192.168.0.11 pour ens38.

```
root@HAProxy1:~# 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: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:6e:ea:cc brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.11/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe6e:eacc/64 scope link
       valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:6e:ea:d6 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.42.137/24 brd 192.168.42.255 scope global dynamic ens37
       valid_lft 1771sec preferred_lft 1771sec
    inet6 fe80::20c:29ff:fe6e:ead6/64 scope link
       valid_lft forever preferred_lft forever
4: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:6e:ea:e0 brd ff:ff:ff:ff:ff:ff
    altname enp2s6
    inet 192.168.0.11/24 brd 192.168.0.255 scope global ens38
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe6e:eae0/64 scope link
       valid_lft forever preferred_lft forever
```

b. Installation et configuration du service HAProxy

Etape 1 :

1. Taper la commande suivante qui permet d'installer le service HAProxy.

Apt install haproxy – y

```
root@HAProxy1:~# apt install haproxy -y
```

2. L'installation du service haproxy s'est effectué et est terminé.

```
root@HAProxy1:~# apt install haproxy -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  liblua5.3-0 libopentracing-c-wrapper0 libopentracing1
Paquets suggérés :
  vim-haproxy haproxy-doc
Les NOUVEAUX paquets suivants seront installés :
  haproxy liblua5.3-0 libopentracing-c-wrapper0 libopentracing1
0 mis à jour, 4 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 2 248 ko dans les archives.
Après cette opération, 5 135 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 liblua5.3-0 amd64 5.3.6-2 [123 kB]
Réception de :2 http://deb.debian.org/debian bookworm/main amd64 libopentracing1 amd64 1.6.0-4 [53,4 kB]
Réception de :3 http://deb.debian.org/debian bookworm/main amd64 libopentracing-c-wrapper0 amd64 1.1.3-3+b1 [29,6 kB]
Réception de :4 http://deb.debian.org/debian bookworm/main amd64 haproxy amd64 2.6.12-1+deb12u1 [2 042 kB]
2 248 ko réceptionnés en 0s (20,5 Mo/s)
Sélection du paquet liblua5.3-0:amd64 précédemment désélectionné.
(Lecture de la base de données... 23249 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../liblua5.3-0_5.3.6-2_amd64.deb ...
Dépaquetage de liblua5.3-0:amd64 (5.3.6-2) ...
Sélection du paquet libopentracing1:amd64 précédemment désélectionné.
Préparation du dépaquetage de .../libopentracing1_1.6.0-4_amd64.deb ...
Dépaquetage de libopentracing1:amd64 (1.6.0-4) ...
Sélection du paquet libopentracing-c-wrapper0:amd64 précédemment désélectionné.
Préparation du dépaquetage de .../libopentracing-c-wrapper0_1.1.3-3+b1_amd64.deb ...
Dépaquetage de libopentracing-c-wrapper0:amd64 (1.1.3-3+b1) ...
Sélection du paquet haproxy précédemment désélectionné.
Préparation du dépaquetage de .../haproxy_2.6.12-1+deb12u1_amd64.deb ...
Dépaquetage de haproxy (2.6.12-1+deb12u1) ...
Paramétrage de liblua5.3-0:amd64 (5.3.6-2) ...
Paramétrage de libopentracing1:amd64 (1.6.0-4) ...
Paramétrage de libopentracing-c-wrapper0:amd64 (1.1.3-3+b1) ...
Paramétrage de haproxy (2.6.12-1+deb12u1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/haproxy.service → /lib/systemd/system/haproxy.service.
Traitement des actions différées (« triggers ») pour libc-bin (2.36-9+deb12u9) ...
```

Etape 2 :

Après l'installation, il faut configurer le service pour assurer le balancement entre les serveurs web du groupe. Ainsi, le fichier de configuration de haproxy est /etc/haproxy/haproxy.cfg. Pour éditer ce dernier, il faut rajouter le terme « nano » pour pouvoir ajouter des lignes au fichier :

nano /etc/haproxy/haproxy.cfg

```
root@SRV-WEB1:~#
root@SRV-WEB1:~#
root@SRV-WEB1:~# nano /etc/haproxy/haproxy.cfg
```

Etape 3 :

Cela ouvrira un fichier dans lequel il faut rajouter les lignes suivantes à la fin :

Configuration du balancement
listen clusterWeb
bind 192.168.0.10 :80
(Futur ip utiliser pour heartbeat)

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

```
#Configuration du balancement
listen clusterWeb
bind 192.168.0.10:80

#Mode d'écoute
mode http

#mode de 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
```

Etape 4 :

1. Taper la commande service haproxy restart ensuite service haproxy status et il faut vérifier qu'il y ait bien écrit Active running et que le voyant soit vert.

```
root@HAProxy1:~# service haproxy restart
root@HAProxy1:~#
Broadcast message from systemd-journald@HAProxy1 (Tue 2025-02-04 23:57:52 CET):

haproxy[1443]: proxy clusterWeb has no server available!


Broadcast message from systemd-journald@HAProxy1 (Tue 2025-02-04 23:57:52 CET):

haproxy[1443]: proxy clusterWeb has no server available!
```

2. Service haproxy status et il faut vérifier qu'il y ait bien écrit Active running et que le voyant soit vert.

```
service haproxy status
● haproxy.service - HAProxy Load Balancer
     Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-02-04 23:57:49 CET; 15s ago
       Docs: man:haproxy(1)
             file:/usr/share/doc/haproxy/configuration.txt.gz
   Main PID: 1440 (haproxy)
      Tasks: 3 (limit: 2273)
     Memory: 41.9M
        CPU: 61ms
     CGroup: /system.slice/haproxy.service
             ├─1440 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
             └─1443 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

févr. 04 23:57:49 HAProxy1 systemd[1]: Started haproxy.service - HAProxy Load Balancer.
févr. 04 23:57:51 HAProxy1 haproxy[1443]: [WARNING]  (1443) : Server clusterWeb/SRV-WEB1 is DOWN, reason: Layer4 timeou
févr. 04 23:57:51 HAProxy1 haproxy[1443]: Server clusterWeb/SRV-WEB1 is DOWN, reason: Layer4 timeout, check duration: 2
févr. 04 23:57:51 HAProxy1 haproxy[1443]: Server clusterWeb/SRV-WEB1 is DOWN, reason: Layer4 timeout, check duration: 2
févr. 04 23:57:52 HAProxy1 haproxy[1443]: [WARNING]  (1443) : Server clusterWeb/SRV-WEB2 is DOWN, reason: Layer4 timeou
févr. 04 23:57:52 HAProxy1 haproxy[1443]: [ALERT]    (1443) : proxy 'clusterWeb' has no server available!
févr. 04 23:57:52 HAProxy1 haproxy[1443]: Server clusterWeb/SRV-WEB2 is DOWN, reason: Layer4 timeout, check duration: 2
févr. 04 23:57:52 HAProxy1 haproxy[1443]: Server clusterWeb/SRV-WEB2 is DOWN, reason: Layer4 timeout, check duration: 2
févr. 04 23:57:52 HAProxy1 haproxy[1443]: proxy clusterWeb has no server available!
févr. 04 23:57:52 HAProxy1 haproxy[1443]: proxy clusterWeb has no server available!
lines 1-23/23 (END)
```

Pour le moment les machines SRV-WEB1 et SRV-WEB2 sont down car ils ne sont pas encore configurés.

(Image une fois les machines actives !)

```
root@HAProxy1:~# service haproxy status
● haproxy.service - HAProxy Load Balancer
     Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-02-05 00:22:53 CET; 54s ago
       Docs: man:haproxy(1)
             file:/usr/share/doc/haproxy/configuration.txt.gz
   Main PID: 1714 (haproxy)
      Tasks: 3 (limit: 2273)
     Memory: 39.9M
        CPU: 69ms
     CGroup: /system.slice/haproxy.service
             ├─1714 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
             └─1717 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

févr. 05 00:22:53 HAProxy1 systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
févr. 05 00:22:53 HAProxy1 haproxy[1714]: [NOTICE]   (1714) : New worker (1717) forked
févr. 05 00:22:53 HAProxy1 haproxy[1714]: [NOTICE]   (1714) : Loading success.
févr. 05 00:22:53 HAProxy1 systemd[1]: Started haproxy.service - HAProxy Load Balancer.
févr. 05 00:23:27 HAProxy1 haproxy[1717]: 192.168.0.100:59162 [05/Feb/2025:00:23:27.758] clusterWeb clusterWeb/SRV-WEB1
févr. 05 00:23:39 HAProxy1 haproxy[1717]: 192.168.0.100:59163 [05/Feb/2025:00:23:39.888] clusterWeb clusterWeb/SRV-WEB2
lines 1-19/19 (END)
```

c. Installation du service Heartbeat sur la machine HAProxy1

Etape 1 :

Allumez les machines, ouvrir le fichier nano /etc/hosts

```
root@HAProxy1:~# nano /etc/hosts
```

Il doit uniquement contenir les lignes suivantes :

192.168.0.11 HAProxy1

192.168.0.12 HAProxy2

172.20.0.21 SRV-WEB1

172.20.0.22 SRV-WEB2

Enregistrez le fichier.

```
 GNU nano 7.2
192.168.0.11    HAProxy1
192.168.0.12    HAProxy2
172.168.0.21    SRV-WEB1
172.168.0.22    SRV-WEB2
```

Etape 2 :

1. Afin de télécharger la fonction « heartbeat » on effectue la commande suivante :

Apt install heartbeat -y

```
root@HAProxy1:~# apt install heartbeat -y
```

2. L'installation du service heartbeat s'est effectué et est terminé.

```
Paramétrage de cluster-glue (1.0.12-21+b2) ...
addgroup : Le groupe « haclient » existe déjà en tant que groupe système. Abandon.
L'utilisateur système « hacluster » existe déjà. Abandon.
Created symlink /etc/systemd/system/multi-user.target.wants/logd.service → /lib/systemd/system/logd.service.
Paramétrage de python3-pexpect (4.8.0-4) ...
Paramétrage de python3-pyasn1-modules (0.2.8-1) ...
Paramétrage de python3-cryptography (38.0.4-3+deb12u1) ...
Paramétrage de python3-requests (2.28.1+dfsg-1) ...
Paramétrage de python3-rsa (4.8-1) ...
Paramétrage de python3-suds (1.1.2-1) ...
Paramétrage de python3-openssl (23.0.0-1) ...
Paramétrage de resource-agents (1:4.12.0-2) ...
resource-agents-deps.target is a disabled or a static unit, not starting it.
Paramétrage de python3-google-auth (1.5.1-3) ...
Paramétrage de python3-botocore (1.29.27+repack-1) ...
Paramétrage de python3-google-auth-httplib2 (0.1.0-3) ...
Paramétrage de pacemaker-resource-agents (2.1.5-1+deb12u1) ...
Paramétrage de heartbeat (1:3.0.6-13+b2) ...
update-rc.d: warning: start and stop actions are no longer supported; falling back to defaults
Created symlink /etc/systemd/system/multi-user.target.wants/heartbeat.service → /lib/systemd/system/heartbeat.service.
Paramétrage de python3-oauth2client (4.1.3-3) ...
Paramétrage de python3-s3transfer (0.6.0-1) ...
Paramétrage de python3-boto3 (1.26.27+dfsg-1) ...
Paramétrage de pacemaker (2.1.5-1+deb12u1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/pacemaker.service → /lib/systemd/system/pacemaker.service.
Paramétrage de python3-googleapi (1.7.12-1) ...
Paramétrage de fence-agents (4.12.1-1) ...
Paramétrage de pacemaker-cli-utils (2.1.5-1+deb12u1) ...
Traitement des actions différées (« triggers ») pour systemd (252.33-1~deb12u1) ...
Traitement des actions différées (« triggers ») pour libc-bin (2.36-9+deb12u9) ...
Traitement des actions différées (« triggers ») pour ca-certificates (20230311) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
```

Etape 3 :

À la suite de l'installation de Heartbeat, un lien a été créé qui est le /etc/heartbeat, pointant vers /etc/ha.d.

On peut le vérifier à l'aide de la commande : ls -l /etc/heartbeat

```
root@HAProxy1:~# ls -l /etc/heartbeat
lrwxrwxrwx 1 root root 4  3 janv.  2023 /etc/heartbeat -> ha.d
```

Dans le dossier /etc/ha.d, nous allons créer 3 fichiers :

- ha.cf : fichier de configuration principal.
- haresources : fichier de configuration des ressources.
- Authkeys : information d'authentification.

Etape 4 :

En commençant par le premier fichier avec la commande :

nano /etc/ha.d/ha.cf

```
root@HAProxy1:~# nano /etc/ha.d/ha.cf
```

Ajoutez les lignes suivantes :

Les fichiers logs de heartbeat (evenements relatifs à heartbeat) :

logfile /var/log/heartbeat

logfacility local0

Intervalle entre deux battements de cœur en seconde

keepalive 5

Temps nécessaire avant de considérer qu'un serveur (nœud) est mort (en seconde)

deadtime 30

Interface d'écoute

bcast ens38

liste des nœuds utilisées pour la HD

node HAProxy1 HAProxy2

comportement si le nœud revient dans le réseau

auto_failback on

```
 GNU nano 7.2                                                    /etc/ha.d/ha.cf
#Les ficihers logs de heartbeat (evenements relatifs à heartbeat)
logfile /var/log/heartbeat
logfacility local0

#intervalle entre deux battements de coeur en seconde
keepalive 5

#Temps nécessaire avant de considérer qu'un serveur (noeud) est mort (en seconde)
deadtime 30

#interface d'écoute
bcast ens38

#Liste des noeuds utilisées pour la HD
node HAProxy1 HAProxy2

#Comportement si le noeud revient dans le réseau
auto_failback on
```

Etape 5 :

1. Ensuite le deuxième fichier avec la commande :

nano /etc/ha.d/haresources

```
root@HAProxy1:~# nano /etc/ha.d/haresources
```

Rajouter la ligne suivante à la suite des deux autres lignes déjà présentes # :

HAProxy1 IPaddr::192.168.0.10/24/ens38 haproxy

```
 GNU nano 7.2                                                    /etc/ha.d/haresources
#active l'interface IP Virtuelle avec comme noeud principal HAProxy1
#Syntaxe :hostname IPaddr::IPvirtuelle/CIDR/interface service
HAProxy1 IPaddr::192.168.0.10/24/ens38 haproxy
```

Enregistrez le fichier.

Etape 6 :

1. Enfin, le dernier fichier en tapant la commande :

nano /etc/ha.d/authkeys

```
root@HAProxy1:~# nano /etc/ha.d/authkeys_
```

```
 GNU nano 7.2
auth 1
1 md5 greta
```

auth 1

1 md5 greta

2. Enregistrez le fichier, il faut aussi sécuriser l'accès à ce dernier en faisant :

chmod 600 /etc/ha.d/authkeys

```
root@HAProxy1:~# chmod 600 /etc/ha.d/authkeys
root@HAProxy1:~# _
```

Etape 7 :

1. Pour vérifier le fonctionnement de l'IP virtuelle, on restart Heartbeat avec la commande « service heartbeat restart ».
On vérifie que Heartbeat est bien lancer avec la commande « service heartbeat status ».

```
root@HAProxy1:~# service heartbeat restart
root@HAProxy1:~# service heartbeat status
● heartbeat.service - Heartbeat High Availability Cluster Communication and Membership
     Loaded: loaded (/lib/systemd/system/heartbeat.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-02-05 21:18:02 CET; 5s ago
       Docs: man:heartbeat(8)
             http://www.linux-ha.org/wiki/Documentation
   Main PID: 3307 (heartbeat)
      Tasks: 4 (limit: 2273)
     Memory: 7.1M
        CPU: 38ms
     CGroup: /system.slice/heartbeat.service
             ├─3307 "heartbeat: master control process"
             ├─3311 "heartbeat: FIFO reader"
             ├─3312 "heartbeat: write: bcast ens38"
             └─3313 "heartbeat: read: bcast ens38"

févr. 05 21:18:06 HAProxy1 heartbeat[3307]: [3307]: info: Link haproxy2:ens38 up.
févr. 05 21:18:06 HAProxy1 heartbeat[3307]: [3307]: info: Status update for node haproxy2: status active
févr. 05 21:18:06 HAProxy1 heartbeat[3315]: [3315]: debug: notify_world: setting SIGCHLD Handler to SIG_DFL
févr. 05 21:18:06 HAProxy1 harc(default)[3322]: info: Running /etc/ha.d//rc.d/status status
févr. 05 21:18:06 HAProxy1 heartbeat[3307]: [3307]: info: Comm_now_up(): updating status to active
févr. 05 21:18:06 HAProxy1 heartbeat[3307]: [3307]: info: Local status now set to: 'active'
févr. 05 21:18:07 HAProxy1 heartbeat[3307]: [3307]: info: remote resource transition completed.
févr. 05 21:18:07 HAProxy1 heartbeat[3307]: [3307]: info: remote resource transition completed.
févr. 05 21:18:07 HAProxy1 heartbeat[3307]: [3307]: info: Local Resource acquisition completed. (none)
févr. 05 21:18:07 HAProxy1 heartbeat[3307]: [3307]: info: haproxy2 wants to go standby [foreign]
root@HAProxy1:~# _
```

```
root@HAProxy1:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    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
    link/ether 00:0c:29:6e:ea:cc brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.11/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe6e:eacc/64 scope link
       valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
    link/ether 00:0c:29:6e:ea:d6 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.42.137/24 brd 192.168.42.255 scope global dynamic
       valid_lft 931sec preferred_lft 931sec
    inet6 fe80::20c:29ff:fe6e:ead6/64 scope link
       valid_lft forever preferred_lft forever
4: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
    link/ether 00:0c:29:6e:ea:e0 brd ff:ff:ff:ff:ff:ff
    altname enp2s6
    inet 192.168.0.11/24 brd 192.168.0.255 scope global ens38
       valid_lft forever preferred_lft forever
    inet 192.168.0.10/24 brd 192.168.0.255 scope global secondary e
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe6e:eae0/64 scope link
       valid_lft forever preferred_lft forever
root@HAProxy1:~# _
```

2. Puis on fait ip a, et on constate l'apparition de l'IP 192.168.0.10 sur la carte ens38.

Etape 8 :

Désormais, il faut supprimer le démarrage automatique du service HAProxy car c'est Heartbeat (haresources) qui s'occupera de démarrer le service.

Commande « update-rc.d haproxy remove

```
root@HAProxy1:~# update-rc.d haproxy remove
```

On éteint les services haproxy et heartbeat

« service haproxy stop »

« service heartbeat stop »

```
root@HAProxy1:~# service haproxy stop
root@HAProxy1:~# service heartbeat stop
```

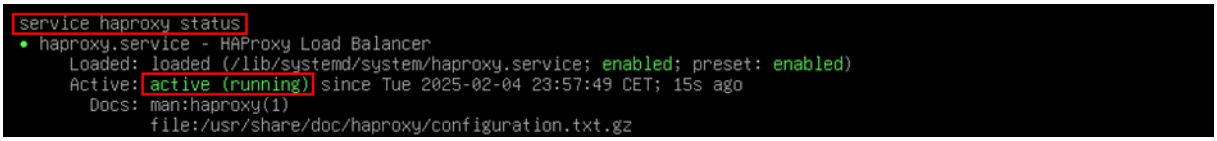
Puis on relance heartbeat

« service heartbeat start »

```
root@HAProxy1:~# service heartbeat start
```

On attend un peu et on vérifie que Haproxy a bien redémarré grâce à Heartbeat.

« service haproxy status »

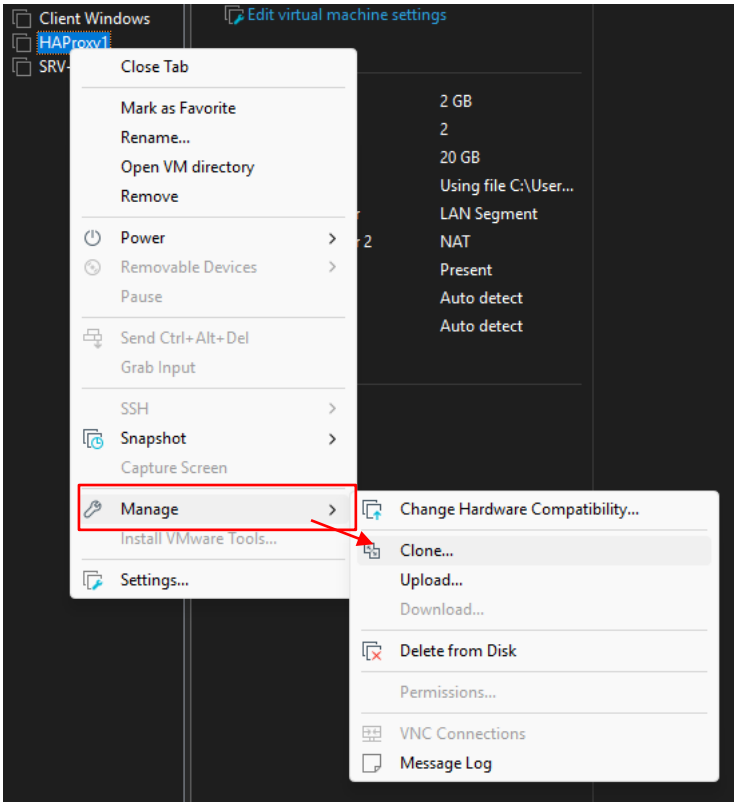


Active : active (running) donc le service a bien redémarré.

d. Clonage HAProxy 1 pour créer HAProxy2

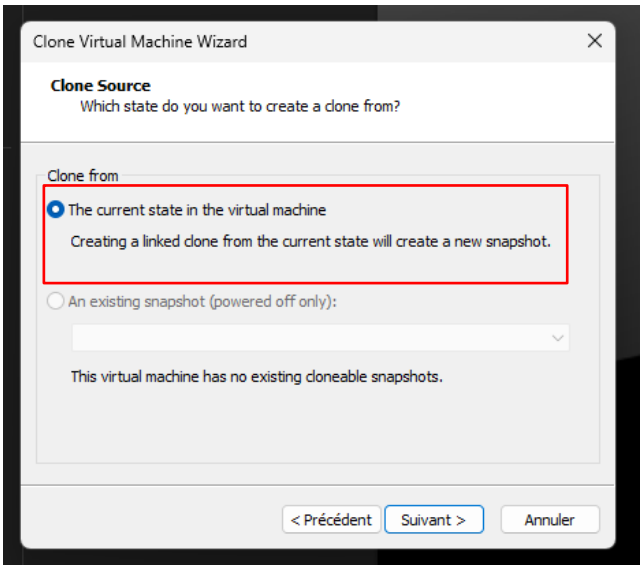
Etape 1 :

Faire un clique droit sur la machine HAProxy1, ensuite aller sur manage puis cliquer sur « Clone… ».



Etape 2 :

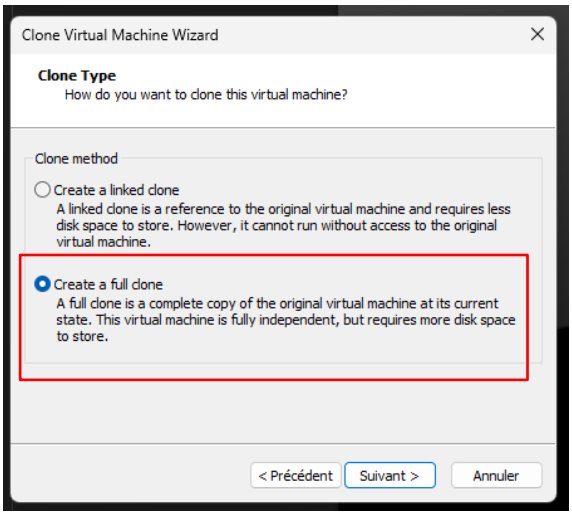
1. Cliquer sur suivant.



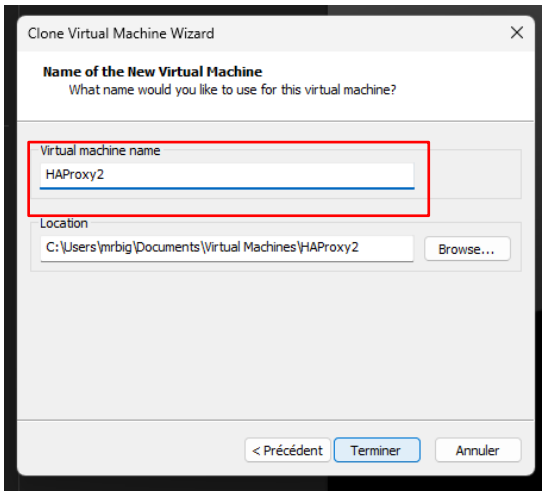
2. Choisir « The current state in the virtual machine » puis faire suivant.

Etape 3 :

Choisir « Create a full clone » et cliquer sur suivant.

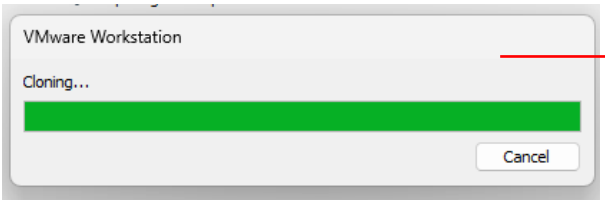


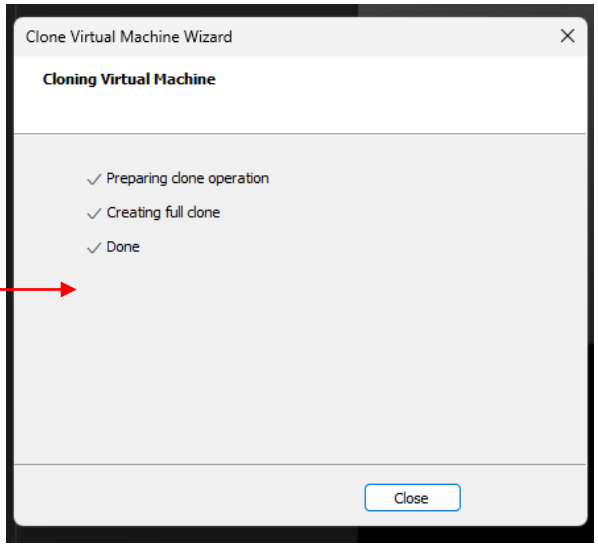
Etape 4 :



1. Mettre le nom de la machine HAProxy2.

2. L'installation s'effectue puis se termine.





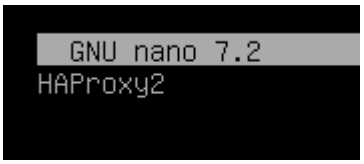
e. *Configuration des adresses IP HAProxy 2*

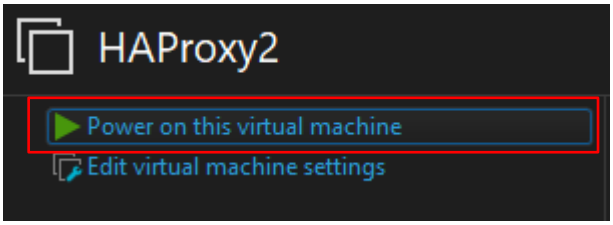
Etape 1 :

Etant donné qu'on a fait une duplication de HAProxy 1 pour le créer, il n'y a pas besoin de faire à nouveau l'installation du service HAProxy.

1. Taper la commande nano /etc/hostname

2. Renommer la machine en HAProxy2.





3. Puis on la redémarre pour qu'elle prenne bien le nouveau nom.

Etape 2 :

1. Allumer la machine, taper la commande nano /etc/network/interfaces.

```
root@HAProxy2:~# nano /etc/network/interfaces
```

2. Modifier les lignes suivantes :
Allow-hotplug ens33
Iface ens33 inet static
Address 172.20.0.22/24

Et on ajoute celle-ci :
Allow-hotplug ens37
Iface ens37 inet dhcp

Allow-hotplug ens38
Iface ens38 inet static
Address 192.168.0.12/24

```
  GNU nano 7.2
# This file describes the network in
# and how to activate them. For more

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 172.20.0.12/24

allow-hotplug ens37
iface ens37 inet dhcp

allow-hotplug ens38
iface ens38 inet static
address 192.168.0.12/24
```

Etape 3 :

1. Faire ifdown ens33, ens37 et ens38.

```
root@HAProxy2:~# ifdown ens33
root@HAProxy2:~# ifdown ens37
Killed old client process
Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens37/00:0c:29:38:3c:9d
Sending on   LPF/ens37/00:0c:29:38:3c:9d
Sending on   Socket/fallback
DHCPRELEASE of 192.168.42.136 on ens37 to 192.168.42.254 port 67
root@HAProxy2:~# ifdown ens38
```

Suite étape 3 :

2. Ensuite faire ifup ens33, ens37 et ens38.

```
root@HAProxy2:~# ifup ens33
root@HAProxy2:~# ifup ens37
Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens37/00:0c:29:38:3c:9d
Sending on   LPF/ens37/00:0c:29:38:3c:9d
Sending on   Socket/fallback
DHCPDISCOVER on ens37 to 255.255.255.255 port 67 interval 6
DHCPOFFER of 192.168.42.136 from 192.168.42.254
DHCPREQUEST for 192.168.42.136 on ens37 to 255.255.255.255 port 67
DHCPACK of 192.168.42.136 from 192.168.42.254
bound to 192.168.42.136 -- renewal in 878 seconds.
root@HAProxy2:~# ifup ens38
```

3. Puis vérifier à l'aide de la commande ip a qu'il y a bien une adresse IP inet Static 172.20.0.12/24 pour ens33 et une adresse IP inet DHCP pour ens37 et une adresse IP 192.168.0.12/24 pour ens38.

```
root@HAProxy2:~# 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: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:38:3c:93 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.12/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe38:3c93/64 scope link
       valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:38:3c:9d brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.42.136/24 brd 192.168.42.255 scope global dynamic ens37
       valid_lft 1786sec preferred_lft 1786sec
    inet6 fe80::20c:29ff:fe38:3c9d/64 scope link
       valid_lft forever preferred_lft forever
4: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:38:3c:a7 brd ff:ff:ff:ff:ff:ff
    altname enp2s6
    inet 192.168.0.12/24 brd 192.168.0.255 scope global ens38
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe38:3ca7/64 scope link
       valid_lft forever preferred_lft forever
root@HAProxy2:~#
```

Etape 4 :

1. Il faut faire une modification, taper la commande nano /etc/haproxy/haproxy.cfg

```
root@HAProxy2:~# nano /etc/haproxy/haproxy.cfg
```

```
#Configuration du balancement
listen clusterWeb
bind 192.168.0.10:80

#Mode d'écoute
mode http

#mode de 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
```

2. Changer l'adresse IP de la configuration du balancement et mettre celle de HAProxy2.

Etape 5 :

Il faut aussi modifier /etc/ha.d/haresources en tapant nano /etc/ha.d.haresources

```
root@HAProxy2:~# nano /etc/ha.d/haresources
```

Rajouter la ligne suivante :
HAProxy2 IPaddr::192.168.0.10/24/ens38 haproxy

```
  GNU nano 7.2
#active l'interface IP Virtuelle avec comme noeud principal HAProxy1
#Syntaxe :hostname IPaddr::IPvirtuelle/CIDR/interface service
HAProxy2 IPaddr::192.168.0.10/24/ens38 haproxy
```

Etape 6 :

1. Pour vérifier le fonctionnement de l'IP virtuel, avec la commande « service heartbeat restart » on redémarre le Hearbeat pour qu'il prenne en compte les modifications
Puis on vérifie que Heartbeat est bien lancer avec la commande « service heartbeat status » en voyant Active running.

```
root@HAProxy2:~# service heartbeat restart
root@HAProxy2:~# service heartbeat status
● heartbeat.service - Heartbeat High Availability Cluster Communication and Membership
     Loaded: loaded (/lib/systemd/system/heartbeat.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-02-05 21:09:26 CET; 9s ago
       Docs: man:heartbeat(8)
             http://www.linux-ha.org/wiki/Documentation
   Main PID: 2447 (heartbeat)
      Tasks: 4 (limit: 2273)
     Memory: 7.0M
        CPU: 281ms
     CGroup: /system.slice/heartbeat.service
             ├─2447 "heartbeat: master control process"
             ├─2451 "heartbeat: FIFO reader"
             ├─2452 "heartbeat: write: bcast ens38"
             └─2453 "heartbeat: read: bcast ens38"

févr. 05 21:09:28 HAProxy2 /usr/lib/ocf/resource.d//heartbeat/IPaddr(IPaddr_192.168.0.10)[2558]: INFO:  Resource is stopped
févr. 05 21:09:28 HAProxy2 ResourceManager(default)[2577]: info: Running /etc/ha.d/resource.d/IPaddr 192.168.0.10/24/ens38 start
févr. 05 21:09:28 HAProxy2 IPaddr(IPaddr_192.168.0.10)[2634]: INFO: Using calculated netmask for 192.168.0.10: 255.255.255.0
févr. 05 21:09:28 HAProxy2 IPaddr(IPaddr_192.168.0.10)[2672]: INFO: eval ifconfig ens38:0 192.168.0.10 netmask 255.255.255.0 broa
févr. 05 21:09:28 HAProxy2 /usr/lib/ocf/resource.d//heartbeat/IPaddr(IPaddr_192.168.0.10)[2707]: INFO:  Success
févr. 05 21:09:28 HAProxy2 ResourceManager(default)[2734]: info: Running /etc/init.d/haproxy  start
févr. 05 21:09:28 HAProxy2 heartbeat[2473]: [2473]: info: local HA resource acquisition completed (standby).
févr. 05 21:09:28 HAProxy2 heartbeat[2447]: [2447]: info: Standby resource acquisition done [foreign].
févr. 05 21:09:28 HAProxy2 heartbeat[2447]: [2447]: info: Initial resource acquisition complete (auto_failback)
févr. 05 21:09:29 HAProxy2 heartbeat[2447]: [2447]: info: remote resource transition completed.
root@HAProxy2:~# _
```

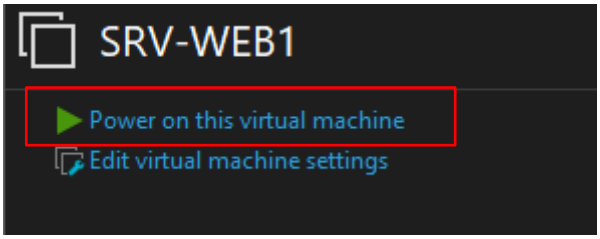
Suite étape 6 :

2. Puis ip a, et on constate l'apparition de l'IP 192.168.0.10 sur la carte ens38.

```
root@HAProxy2:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
    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
    link/ether 00:0c:29:38:3c:93 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.12/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe38:3c93/64 scope link
       valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
    link/ether 00:0c:29:38:3c:9d brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.42.136/24 brd 192.168.42.255 scope global dynamic ens37
       valid_lft 1615sec preferred_lft 1615sec
    inet6 fe80::20c:29ff:fe38:3c9d/64 scope link
       valid_lft forever preferred_lft forever
4: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
    link/ether 00:0c:29:38:3c:a7 brd ff:ff:ff:ff:ff:ff
    altname enp2s6
    inet 192.168.0.12/24 brd 192.168.0.255 scope global ens38
       valid_lft forever preferred_lft forever
    inet 192.168.0.10/24 brd 192.168.0.255 scope global secondary ens38:0
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe38:3ca7/64 scope link
       valid_lft forever preferred_lft forever
root@HAProxy2:~# _
```

3. ***Mise en place d'Apache2* (php, template, boostrap) *pour les machines SRV-WEB1 et SRV-WEB2 DEBIAN***
    a. ***Installation et configuration du service apache2 et installation du site thegrill (template, boostrap)***

Etape 1 :



1. Allumez la machine virtuelle

2. Puis taper la commande suivante qui permet d'installer le service apache2 :

apt install apache2 -y

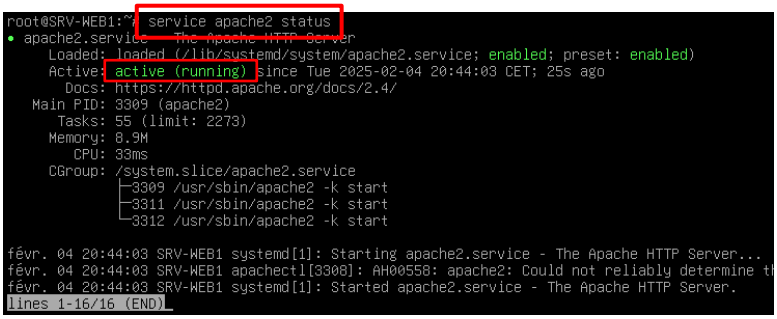
```
root@SRV-WEB1:~# apt install apache2 -y
```

```
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.targe
Created symlink /etc/systemd/system/multi-user.targe
Traitement des actions différées (« triggers ») pour
Traitement des actions différées (« triggers ») pour
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
```

3. L'installation s'est finie sans encombre.

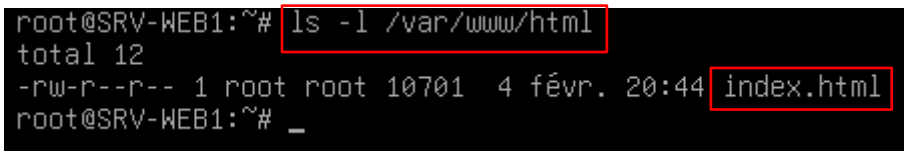
Etape 2 :

Ensuite il faut vérifier le statut en faisant la commande :
service apache2 status

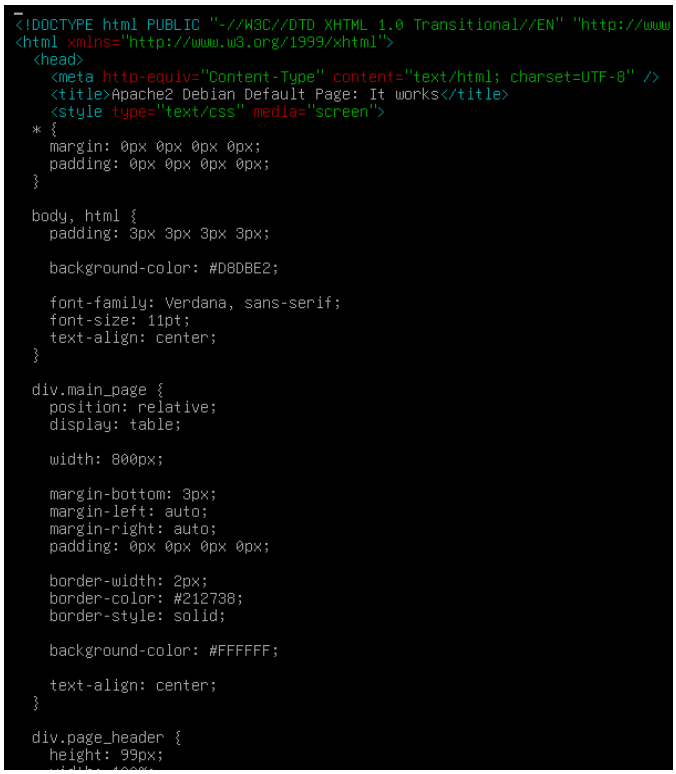


Etape 3 :

Vérifier le documentroot d'apache en tapant la commande
ls -l /var/www/html qui affichera un dossier index.html.



Pour consulter son contenu, c'est la commande
nano /var/www/html/index.html

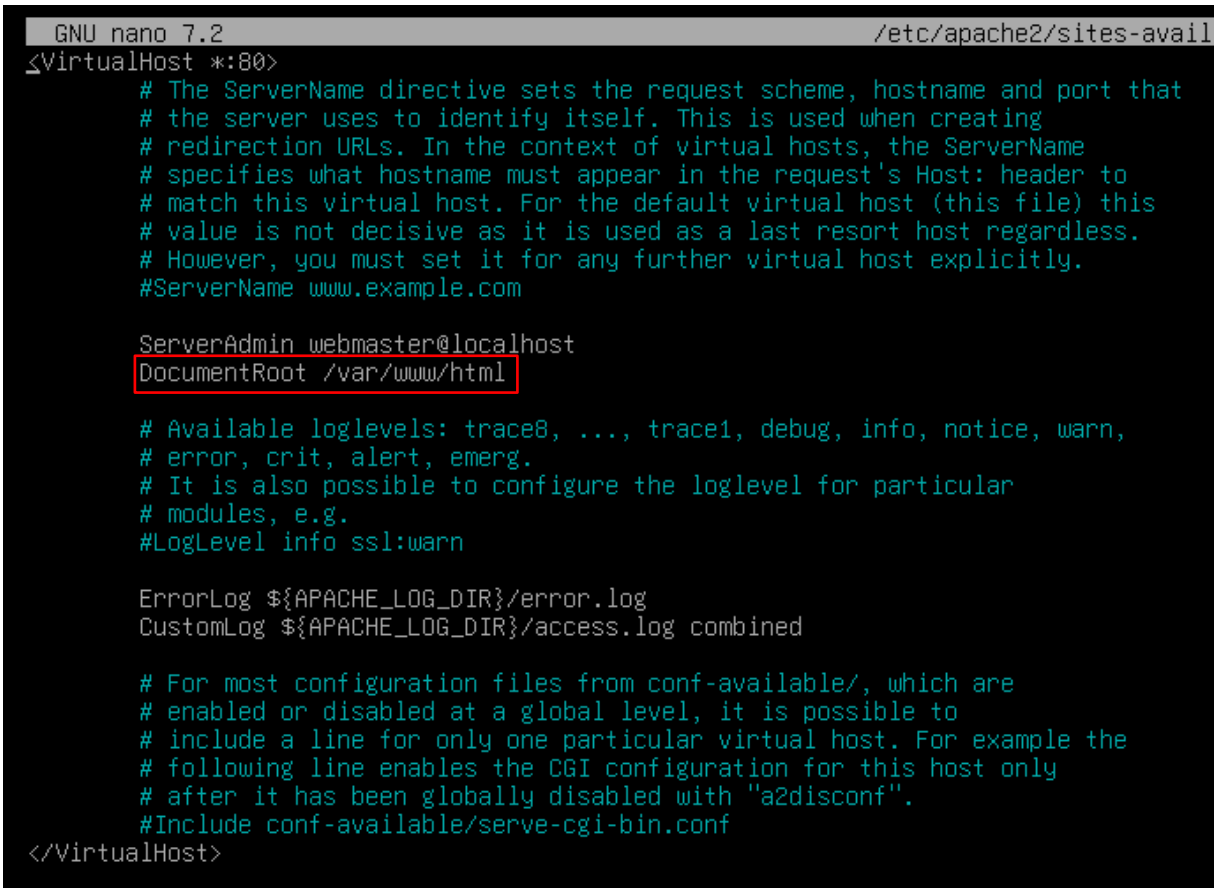


Etape 4 :

Le documentroot d'apache est renseigné dans le fichier
nano /etc/apache2/sites-available/000-default.conf

```
root@SRV-WEB1:~# nano /etc/apache2/sites-available/000-default.conf
```

Vérifier que le documentroot s'y trouve.



Etape 5 :

1. Par la suite, taper la commande apt install wget unzip -y
(wget est un diminutif de we get permettant de télécharger des fichiers via internet).
(unzip permet de décompresser des fichiers zip).

```
root@SRV-WEB1:~# apt install wget unzip -y
```

Suite étape 5 :

2. L'installation démarre des nouveaux paquets.

```
root@SRV-WEB1:~# apt install wget unzip -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Paquets suggérés :
  zip
Les NOUVEAUX paquets suivants seront installés :
  unzip wget
0 mis à jour, 2 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 1 150 ko dans les archives.
Après cette opération, 4 080 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 wget amd64 1.21.3-1+b2 [984 kB]
Réception de :2 http://deb.debian.org/debian bookworm/main amd64 unzip amd64 6.0-28 [166 kB]
1 150 ko réceptionnés en 0s (14,7 Mo/s)
Sélection du paquet wget précédemment désélectionné.
(Lecture de la base de données... 26572 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../wget_1.21.3-1+b2_amd64.deb ...
Dépaquetage de wget (1.21.3-1+b2) ...
Sélection du paquet unzip précédemment désélectionné.
Préparation du dépaquetage de .../unzip_6.0-28_amd64.deb ...
Dépaquetage de unzip (6.0-28) ...
Paramétrage de wget (1.21.3-1+b2) ...
Paramétrage de unzip (6.0-28) ...
```

Etape 6 :

1. Taper la commande :

cd /var/www/html

```
root@SRV-WEB1:~# cd /var/www/html_
```

2. Ensuite celle-ci :

Wget https://github.com/technext/thegrill/archive/master.zip

```
root@SRV-WEB1:/var/www/html# wget https://github.com/technext/thegrill/archive/master.zip
--2025-02-04 20:49:41--  https://github.com/technext/thegrill/archive/master.zip
Résolution de github.com (github.com)… 140.82.121.4
Connexion à github.com (github.com)|140.82.121.4|:443… connecté.
requête HTTP transmise, en attente de la réponse… 302 Found
Emplacement : https://codeload.github.com/technext/thegrill/zip/refs/heads/master [suivant]
--2025-02-04 20:49:42--  https://codeload.github.com/technext/thegrill/zip/refs/heads/master
Résolution de codeload.github.com (codeload.github.com)… 140.82.121.10
Connexion à codeload.github.com (codeload.github.com)|140.82.121.10|:443… connecté.
requête HTTP transmise, en attente de la réponse… 200 OK
Taille : non indiqué [application/zip]
Sauvegarde en : « master.zip »

master.zip                              [ <=>

2025-02-04 20:49:42 (22,2 MB/s) - « master.zip » sauvegardé [4534581]
```

3. Faire ls -l pour vérifier que le master.zip soit ajouté :

```
root@SRV-WEB1:/var/www/html# ls -l
total 4444
-rw-r--r-- 1 root root   10701  4 févr. 20:44 index.html
-rw-r--r-- 1 root root 4534581  4 févr. 20:49 master.zip
root@SRV-WEB1:/var/www/html# _
```

Suite étape 6 :

4. Taper la commande :
Unzip master.zip

```
root@SRV-WEB1:/var/www/html# unzip master.zip
```

5. Refaire ls -l
On voit bien la présence d'un new folder « thegrill-master » résultat de la décompression du fichier master.zip.

```
root@SRV-WEB1:/var/www/html# ls -l
total 4448
-rw-r--r-- 1 root root   10701  4 févr. 20:44 index.html
-rw-r--r-- 1 root root 4534581  4 févr. 20:49 master.zip
drwxr-xr-x 3 root root    4096 21 août   2019 thegrill-master
root@SRV-WEB1:/var/www/html# _
```

6. Taper cette commande :
Rm (remove) master.zip
Cette commande permet de supprimer le master.zip.
Vérifier en tapant ls -l

```
root@SRV-WEB1:/var/www/html# rm master.zip
root@SRV-WEB1:/var/www/html# ls -l
total 16
-rw-r--r-- 1 root root 10701  4 févr. 20:44 index.html
drwxr-xr-x 3 root root  4096 21 août   2019 thegrill-master
root@SRV-WEB1:/var/www/html#
```

Le fait d'avoir ajouté 2019 thegrill-master va permettre de consulter le site internet en mettant l'adresse IP sur une page internet.

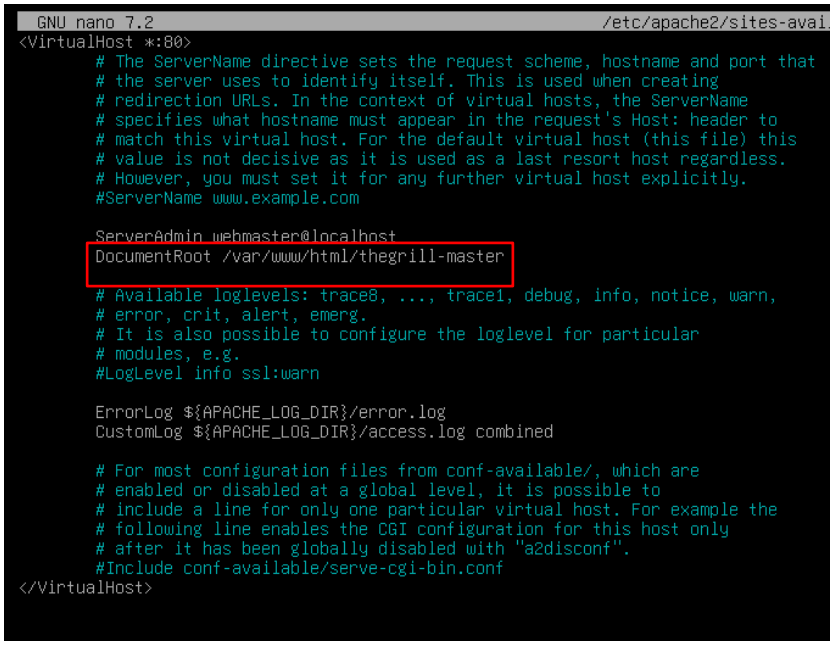
Etape 7 :

1. A la suite de cela, taper la commande :
nano /etc/apache2/sites-available/000-default.conf

```
root@SRV-WEB1:~# nano /etc/apache2/sites-available/000-default.conf_
```

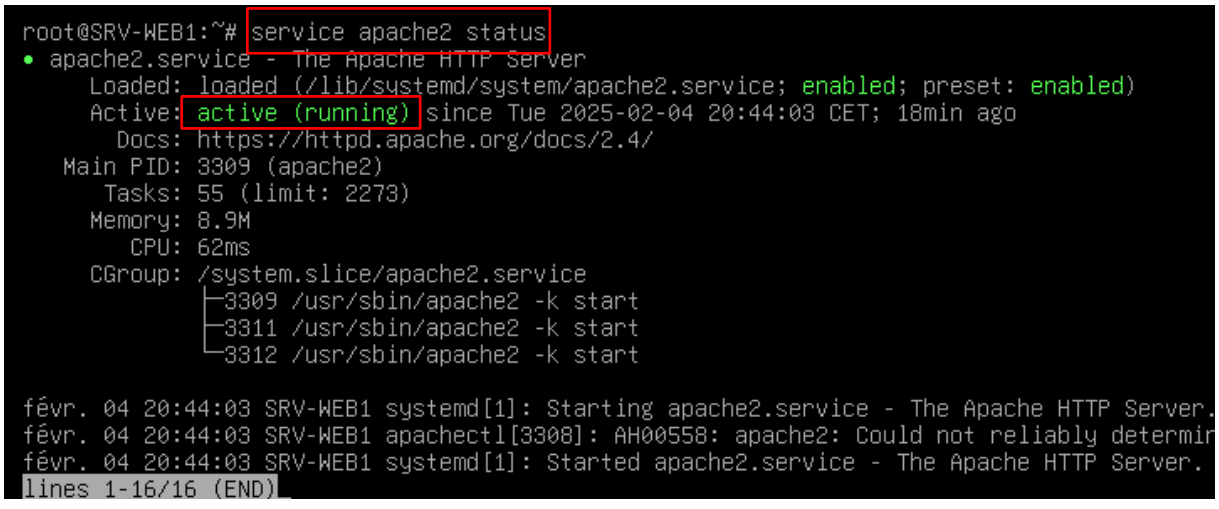
Suite étape 7 :

2. Modifier le documentroot d'apache en ajoutant /thegrill-master après le /var/www/html.



Etape 8 :

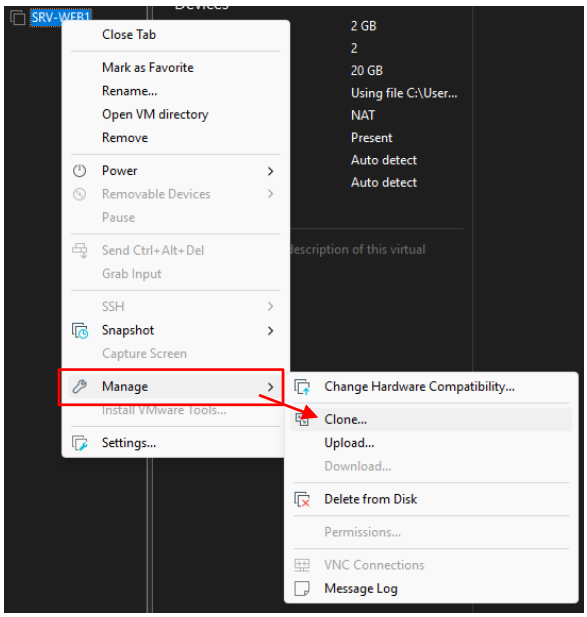
Taper service apache2 restart et ensuite service apache2 status et vérifier qu'il y ait bien écrit Active running que que le voyant soit vert.



b. *Clonage SRV-WEB1 pour créer SRV-WEB2*

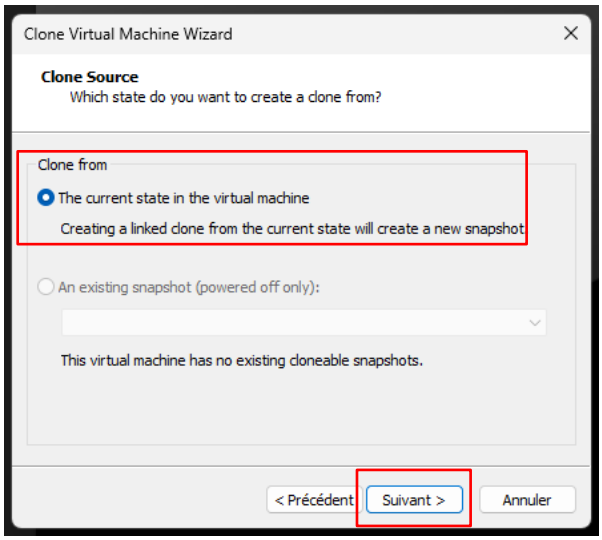
Etape 1 :

Faire un clique droit sur la machine SRV-WEB1, ensuite aller sur manage puis cliquer sur « Clone... ».



Etape 2 :

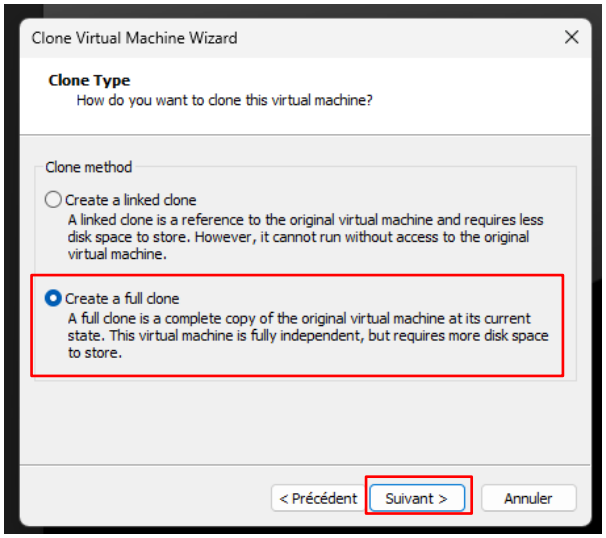
1. Cliquer sur suivant.



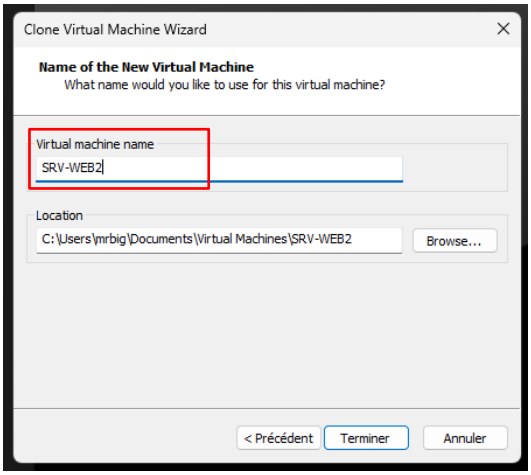
2. Choisir « The current state in the virtual machine » puis faire suivant.

Etape 3 :

Choisir « Create a full clone » et cliquer sur suivant.

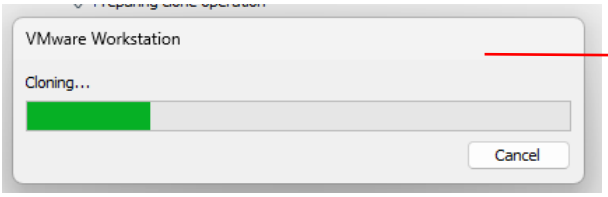


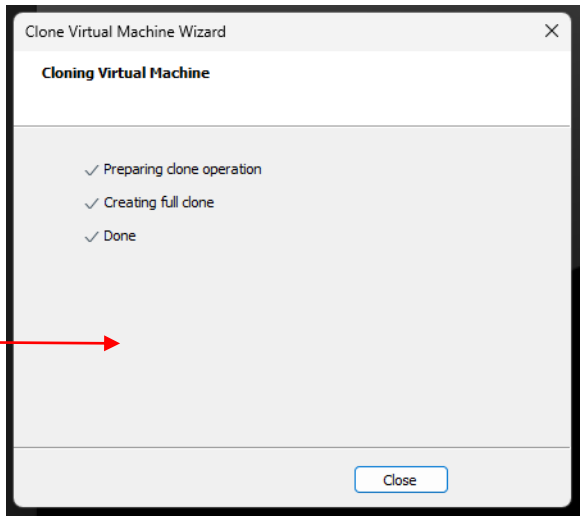
Etape 4 :



1. Mettre le nom de la machine SRV-WEB2.

2. L'installation s'effectue puis se termine.





Etape 5 :

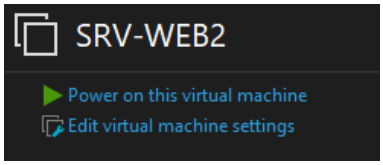
Etant donné qu'on a fait une duplication de SRV-WEB1 pour le créer, il n'y a pas besoin de faire à nouveau les installations et configurations.

```
root@SRV-WEB1:~# nano /etc/hostname
```

1. Taper la commande nano /etc/hostname

2. Renommer la machine en SRV-WEB2

```
 GNU nano 7.2
SRV-WEB2_
```



3. Puis on la redémarre pour qu'elle prenne bien le nouveau nom.

Etape 6 :

Afin de différencier les deux machines WEB, on démarre SRV-WEB1 SRV-WEB2 pour ajouter respectivement 1 et 2 après le titre THE GRILL des SRV-WEB 1 et 2 dans le fichier /var/www/html/thegrill-master/index.html

SRV-WEB 1

```
root@SRV-WEB1:~# nano /var/www/html/thegrill-master/index.html_
```

```
</head>

<body id="page-top" class="regular-navigation">


    <div class="master-wrapper">

        <div class="preloader">
            <div class="preloader-img">
                <span class="loading-animation animate-flicker"><img src="assets/img/loading.GIF" alt="loading" /></span>
            </div>
        </div>

        <!-- Navigation -->
        <nav class="navbar navbar-default navbar-fixed-top fadeInDown" data-wow-delay="0.2s">
            <div class="container-fluid">
                <!-- Brand and toggle get grouped for better mobile display -->
                <div class="navbar-header page-scroll">
                    <button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#main-navigation">
                        <span class="sr-only">Toggle navigation</span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                    </button>
                    <a class="navbar-brand smoothie" href="index.html">THE <span class="theme-accent-color">GRILL 1</span></a>
                </div>
```

Suite étape 6 :

```
root@SRV-WEB2:~# nano /var/www/html/thegrill-master/index.html
```

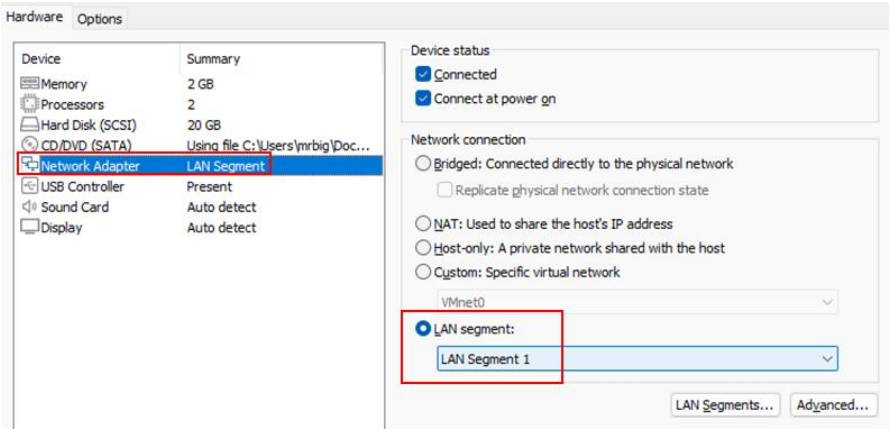
```
</head>

<body id="page-top" class="regular-navigation">


    <div class="master-wrapper">

        <div class="preloader">
            <div class="preloader-img">
                <span class="loading-animation animate-flicker"><img src="assets/img/loading.GIF" alt="loading" /></span>
            </div>
        </div>

        <!-- Navigation -->
        <nav class="navbar navbar-default navbar-fixed-top fadeInDown" data-wow-delay="0.2s">
            <div class="container-fluid">
                <!-- Brand and toggle get grouped for better mobile display -->
                <div class="navbar-header page-scroll">
                    <button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#main-navigation">
                        <span class="sr-only">Toggle navigation</span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                        <span class="icon-bar"></span>
                    </button>
                    <a class="navbar-brand smoothie" href="index.html">THE <span class="theme-accent-color">GRILL 2</span></a>
                </div>
```

c. *Configuration de l'adresse IP SRV-WEB1*

Etape 1 :

Il faut se rendre dans les paramètres de la machine virtuelle SRV-WEB1, en cliquant sur « settings ».

Ensuite modifier le network adapter NAT pour le mettre sur le LAN segment « LAN segment 1 »



Etape 2 :

1. Allumer la machine, taper la commande nano /etc/network/interfaces

```
root@SRV-WEB1:~# nano /etc/network/interfaces_
```

```
  GNU nano 7.2
# This file describes the network ir
# and how to activate them. For more

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 172.20.0.21/24
```

2. Modifier les lignes suivantes :
Allow-hotplug ens33
Iface ens33 inet static
Address 172.20.0.21/24

Etape 3 :

1. Faire ifdown ens33 juste après ifup ens 33.

```
root@SRV-WEB1:~# ifdown ens33
root@SRV-WEB1:~# ifup ens33
```

2. Puis vérifier à l'aide de la commande ip a qu'il y a bien une adresse IP inet Static 172.20.0.21/24 pour ens33.

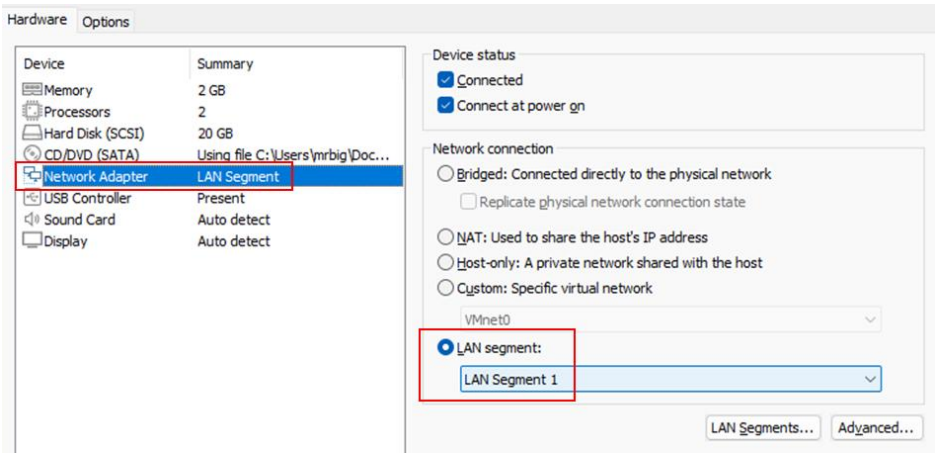
```
root@SRV-WEB1:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:53:8a:e8 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.21/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe53:8ae8/64 scope link
       valid_lft forever preferred_lft forever
root@SRV-WEB1:~# _
```

d. Configuration de l'adresse IP SRV-WEB2

Etape 1 :

Il faut se rendre dans les paramètres de la machine virtuelle SRV-WEB1, en cliquant sur « settings ».

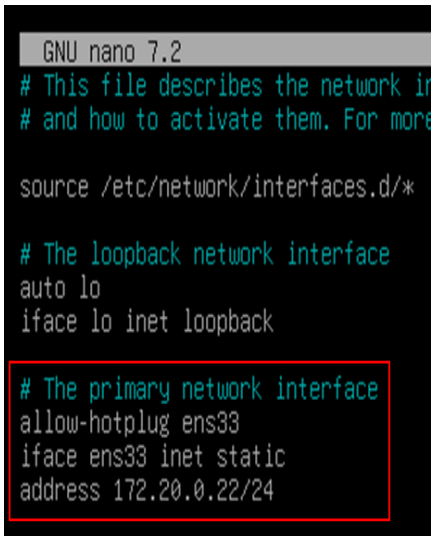
Ensuite modifier le network adapter NAT pour le mettre sur le LAN segment « LAN segment 1 »



Etape 2 :

1. Allumer la machine, taper la commande nano /etc/network/interfaces.

```
root@SRV-WEB2:~# nano /etc/network/interfaces
```



2. Modifier les lignes suivantes :
Allow-hotplug ens33
Iface ens33 inet static
Address 172.20.0.22/24

Etape 3 :

1. Faire ifdown puis ifup ens33

```
root@SRV-WEB2:~# ifdown ens33
root@SRV-WEB2:~# ifup ens33
```

2. Puis vérifier à l'aide de la commande ip a qu'il y a bien une adresse IP inet Static 172.20.0.22/24 pour ens33.

```
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: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:b2:cb:43 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.22/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:feb2:cb43/64 scope link
       valid_lft forever preferred_lft forever
root@SRV-WEB2:~#
```

4. *Test*

a. *Vérification de la connectivité (ping entre les machines)*

Etape 1 :

1. Ensuite allumez les machines, taper la commande ip a pour vérifier les configurations des adresses IP.

2. HAProxy1 :
ens33 172.20.0.11/24
ens37 dhcp
ens38 192.168.0.11 et 192.168.0.10

```
root@HAProxy1:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
    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
    link/ether 00:0c:29:6e:ea:cc brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.11/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe6e:eacc/64 scope link
       valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
    link/ether 00:0c:29:6e:ea:d6 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.42.137/24 brd 192.168.42.255 scope global dynamic ens37
       valid_lft 1502sec preferred_lft 1502sec
    inet6 fe80::20c:29ff:fe6e:ead6/64 scope link
       valid_lft forever preferred_lft forever
4: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
    link/ether 00:0c:29:6e:ea:e0 brd ff:ff:ff:ff:ff:ff
    altname enp2s6
    inet 192.168.0.11/24 brd 192.168.0.255 scope global ens38
       valid_lft forever preferred_lft forever
    inet 192.168.0.10/24 brd 192.168.0.255 scope global secondary ens38:0
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe6e:eae0/64 scope link
       valid_lft forever preferred_lft forever
root@HAProxy1:~#
```

Suite étape 1 :

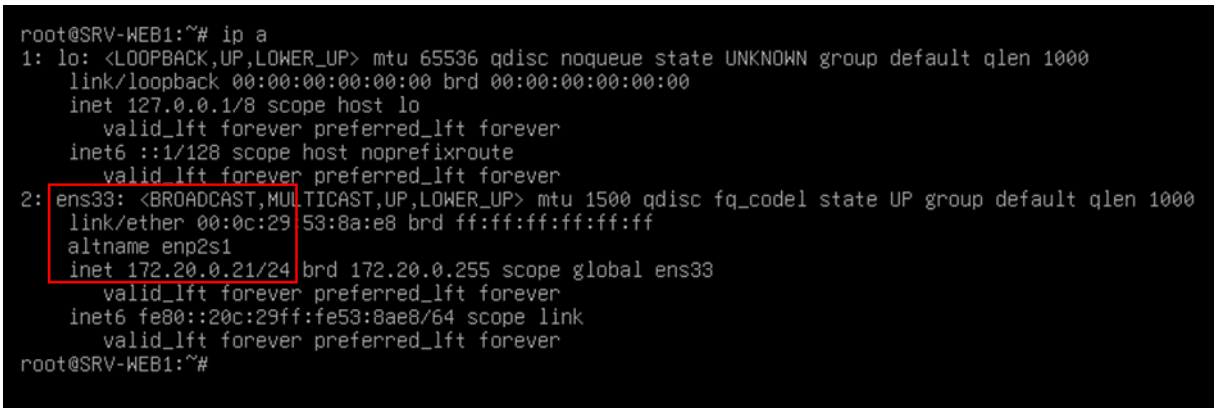
3. HAProxy2 :
ens33 172.20.0.12/24
ens37 dhcp
ens38 192.168.0.12 et 192.168.0.10

```
root@HAProxy2:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
    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
    link/ether 00:0c:29:38:3c:93 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.20.0.12/24 brd 172.20.0.255 scope global ens33
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe38:3c93/64 scope link
       valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
    link/ether 00:0c:29:38:3c:9d brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.42.136/24 brd 192.168.42.255 scope global dynamic ens37
       valid_lft 1762sec preferred_lft 1762sec
    inet6 fe80::20c:29ff:fe38:3c9d/64 scope link
       valid_lft forever preferred_lft forever
4: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
    link/ether 00:0c:29:38:3c:a7 brd ff:ff:ff:ff:ff:ff
    altname enp2s6
    inet 192.168.0.12/24 brd 192.168.0.255 scope global ens38
       valid_lft forever preferred_lft forever
    inet 192.168.0.10/24 brd 192.168.0.255 scope global secondary ens38:0
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe38:3ca7/64 scope link
       valid_lft forever preferred_lft forever
root@HAProxy2:~#
```

Suite étape 1 :

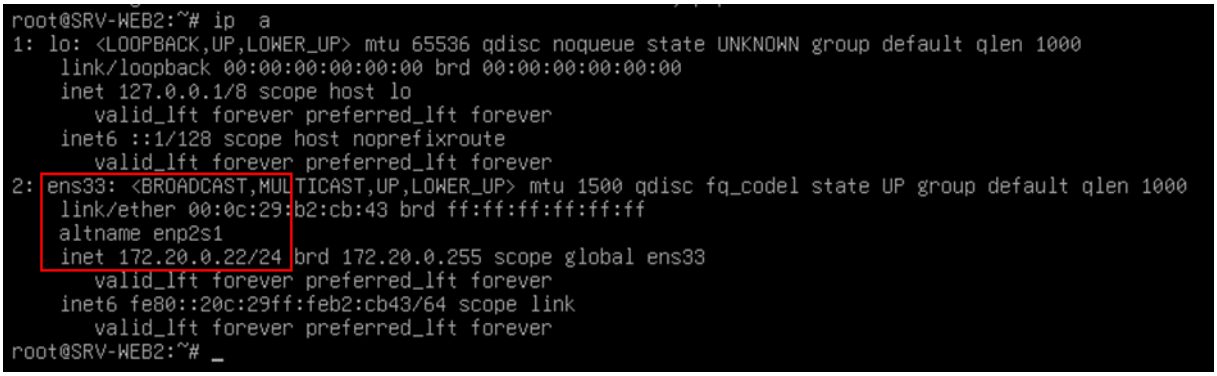
4. SRV-WEB1 :

ens33 172.20.0.21/24



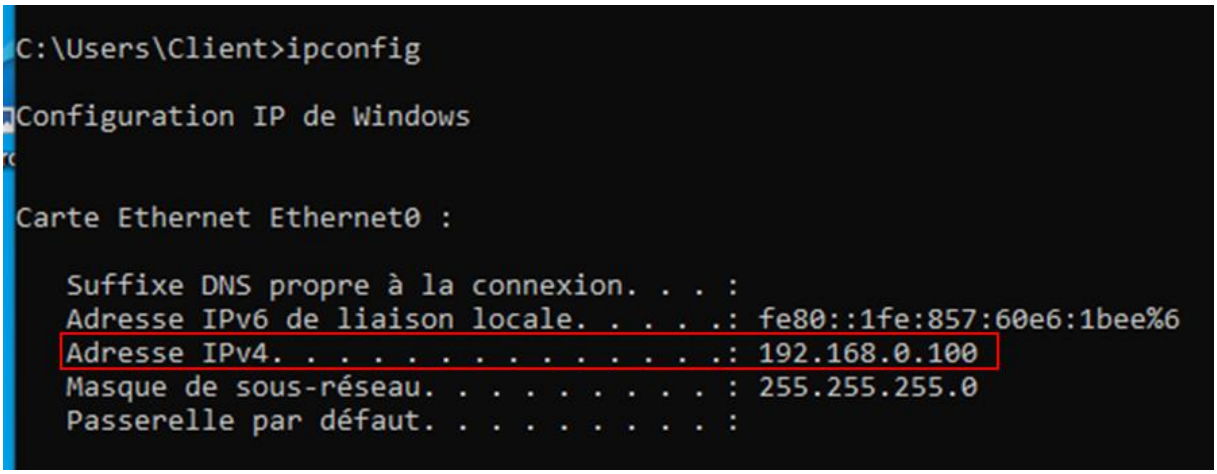
5. SRV-WEB2 :

ens33 172.20.0.22/24



6. Client Windows :

Adresse IPv4 192.168.0.100



Etape 2 :

Faire un ping entre les 4 machines ainsi que Heartbeat.

1. Depuis HAProxy1 :

Ping 172.20.0.21 (SRV-WEB1)

Ping 172.20.0.22 (SRV-WEB2)

Ping 172.20.0.12 (HAProxy2)

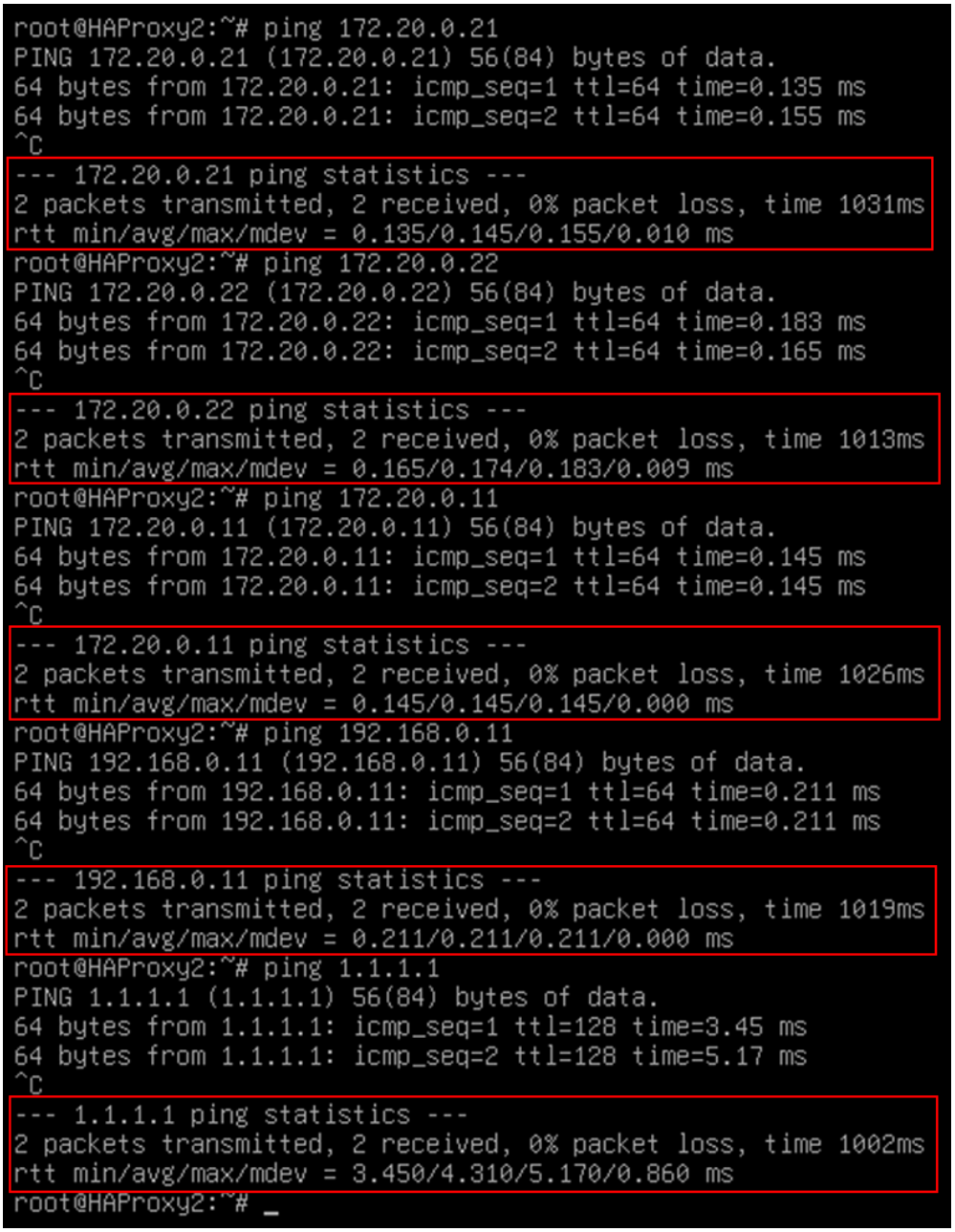
Ping 192.168.0.12 (HAProxy2)

Ping 1.1.1.1 (DNS)

```
root@HAProxy1:~# ping 172.20.0.21
PING 172.20.0.21 (172.20.0.21) 56(84) bytes of data.
64 bytes from 172.20.0.21: icmp_seq=1 ttl=64 time=0.151 ms
64 bytes from 172.20.0.21: icmp_seq=2 ttl=64 time=0.152 ms
^C
--- 172.20.0.21 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1019ms
rtt min/avg/max/mdev = 0.151/0.151/0.152/0.000 ms
root@HAProxy1:~# ping 172.20.0.22
PING 172.20.0.22 (172.20.0.22) 56(84) bytes of data.
64 bytes from 172.20.0.22: icmp_seq=1 ttl=64 time=0.157 ms
64 bytes from 172.20.0.22: icmp_seq=2 ttl=64 time=0.221 ms
^C
--- 172.20.0.22 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1015ms
rtt min/avg/max/mdev = 0.157/0.189/0.221/0.032 ms
root@HAProxy1:~# ping 172.20.0.12
PING 172.20.0.12 (172.20.0.12) 56(84) bytes of data.
64 bytes from 172.20.0.12: icmp_seq=1 ttl=64 time=0.155 ms
64 bytes from 172.20.0.12: icmp_seq=2 ttl=64 time=0.134 ms
^C
--- 172.20.0.12 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1027ms
rtt min/avg/max/mdev = 0.134/0.144/0.155/0.010 ms
root@HAProxy1:~# ping 192.168.0.12
PING 192.168.0.12 (192.168.0.12) 56(84) bytes of data.
64 bytes from 192.168.0.12: icmp_seq=1 ttl=64 time=0.345 ms
64 bytes from 192.168.0.12: icmp_seq=2 ttl=64 time=0.212 ms
^C
--- 192.168.0.12 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.212/0.278/0.345/0.066 ms
root@HAProxy1:~# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=128 time=3.38 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=128 time=3.25 ms
^C
--- 1.1.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 3.247/3.313/3.379/0.066 ms
root@HAProxy1:~# _
```

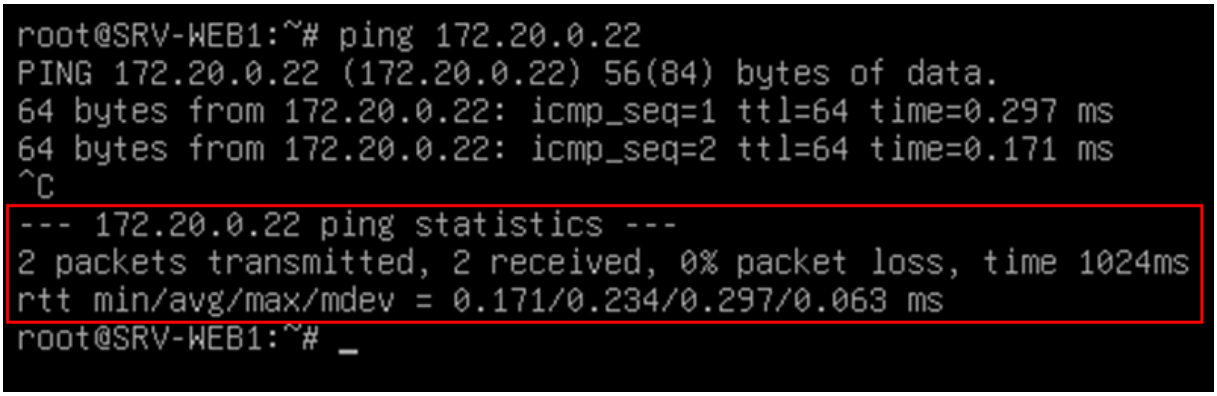
Suite étape 2 :

2. Depuis HAProxy2 :
Ping 172.20.0.21 (SRV-WEB1)
Ping 172.20.0.22 (SRV-WEB2)
Ping 172.20.0.11 (HAProxy1)
Ping 192.168.0.11 (HAProxy1)
Ping 1.1.1.1 (DNS)

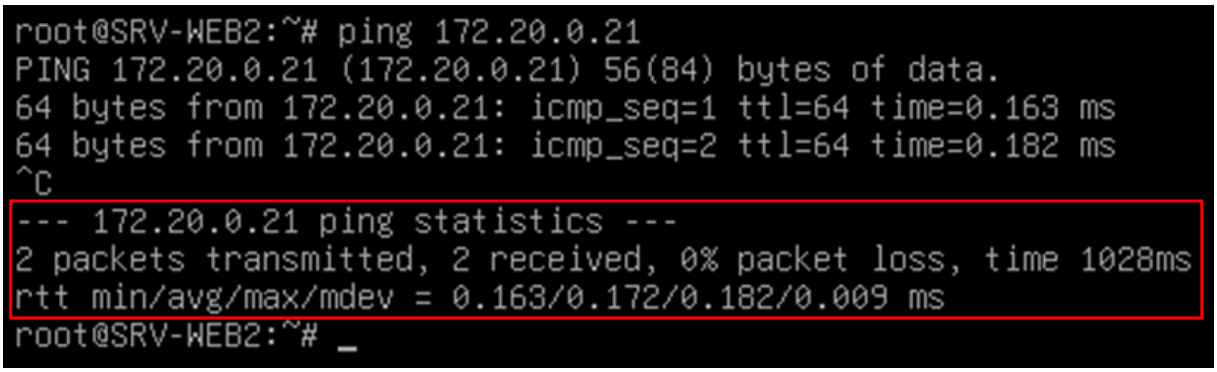


Suite étape 2 :

3. Depuis SRV-WEB1 :
Ping 172.20.0.22 (SRV-WEB2)



4. Depuis SRV-WEB 2 :
Ping 172.20.0.21 (SRV-WEB1)



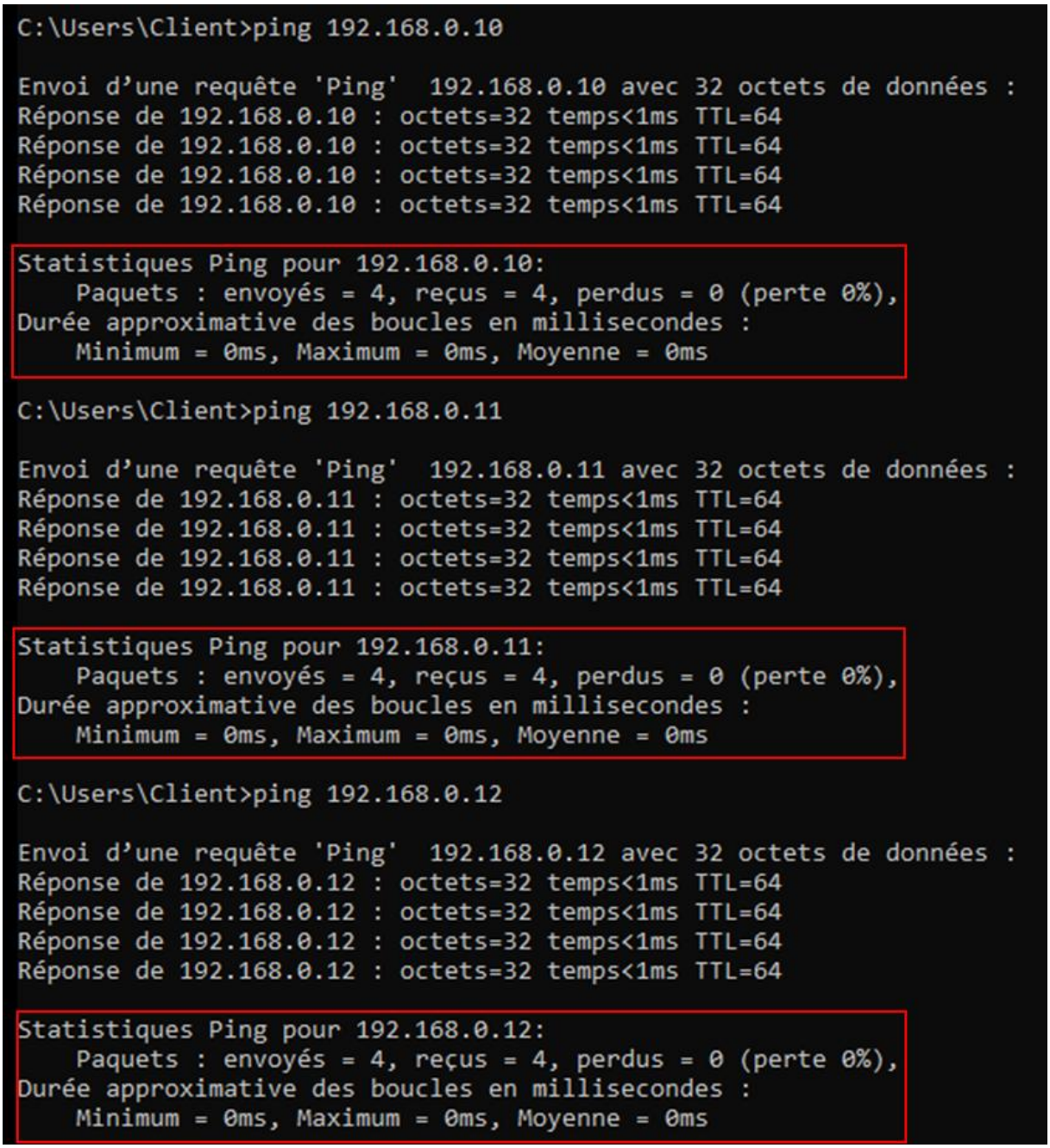
Suite étape 2 :

5. Client Windows :

Ping 192.168.0.10 (Heartbeat)

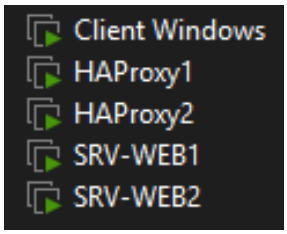
Ping 192.168.0.11 (HAProxy1)

Ping 192.168.0.12 (HAProxy2)



b. Vérification du fonctionnement de HAProxy

Etape 1 :



1. Pour vérifier le fonctionnement de HAProxy.
Il faut mettre en route toutes les machines.

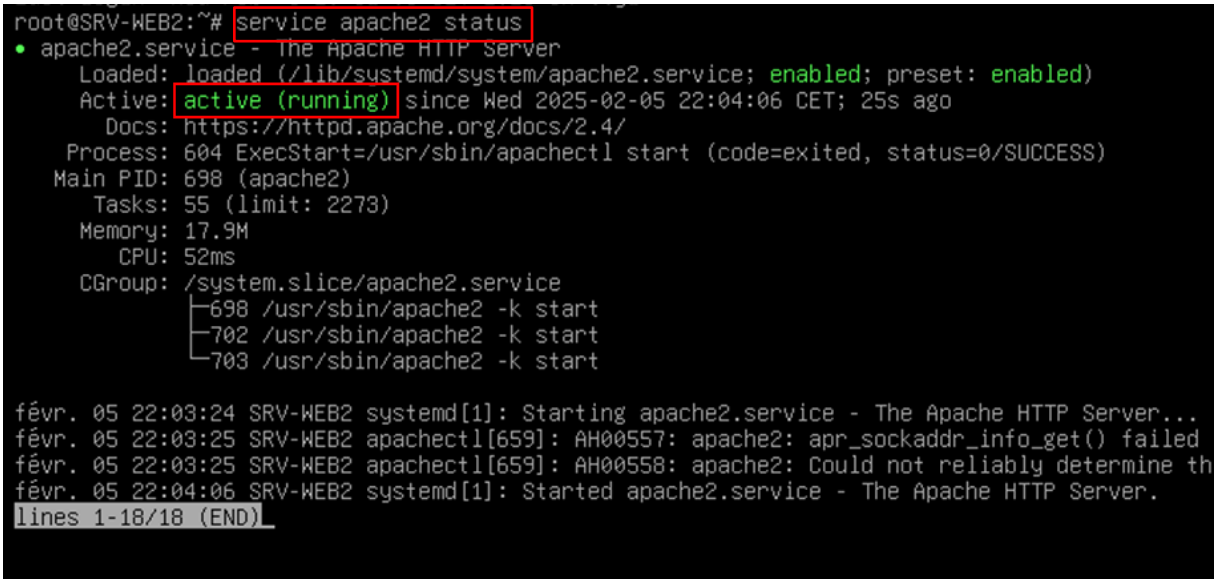
2. Vérifier que le service apache2 soit allumé sur les SRV-WEB1 et SRV-WEB2 en faisant :
Service apache2 status
Il faut qu'il y ait écrit Active running.

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 Wed 2025-02-05 22:04:08 CET; 56s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 517 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 675 (apache2)
      Tasks: 55 (limit: 2273)
     Memory: 11.9M
        CPU: 58ms
     CGroup: /system.slice/apache2.service
             ├─675 /usr/sbin/apache2 -k start
             ├─682 /usr/sbin/apache2 -k start
             └─683 /usr/sbin/apache2 -k start

févr. 05 22:03:27 SRV-WEB1 systemd[1]: Starting apache2.service - The Apache HTTP Server...
févr. 05 22:04:08 SRV-WEB1 apachectl[628]: AH00558: apache2: Could not reliably determine th
févr. 05 22:04:08 SRV-WEB1 systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-17/17 (END)
```

Suite étape 1 :

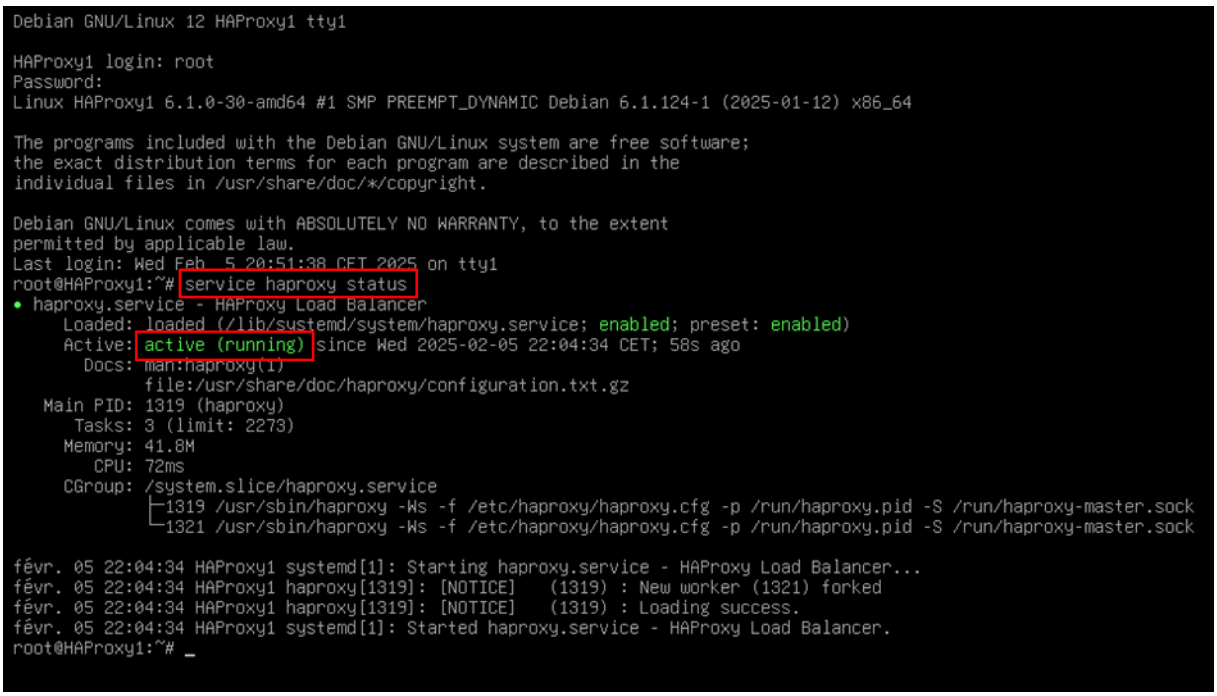
SRV-WEB2



Etape 2 :

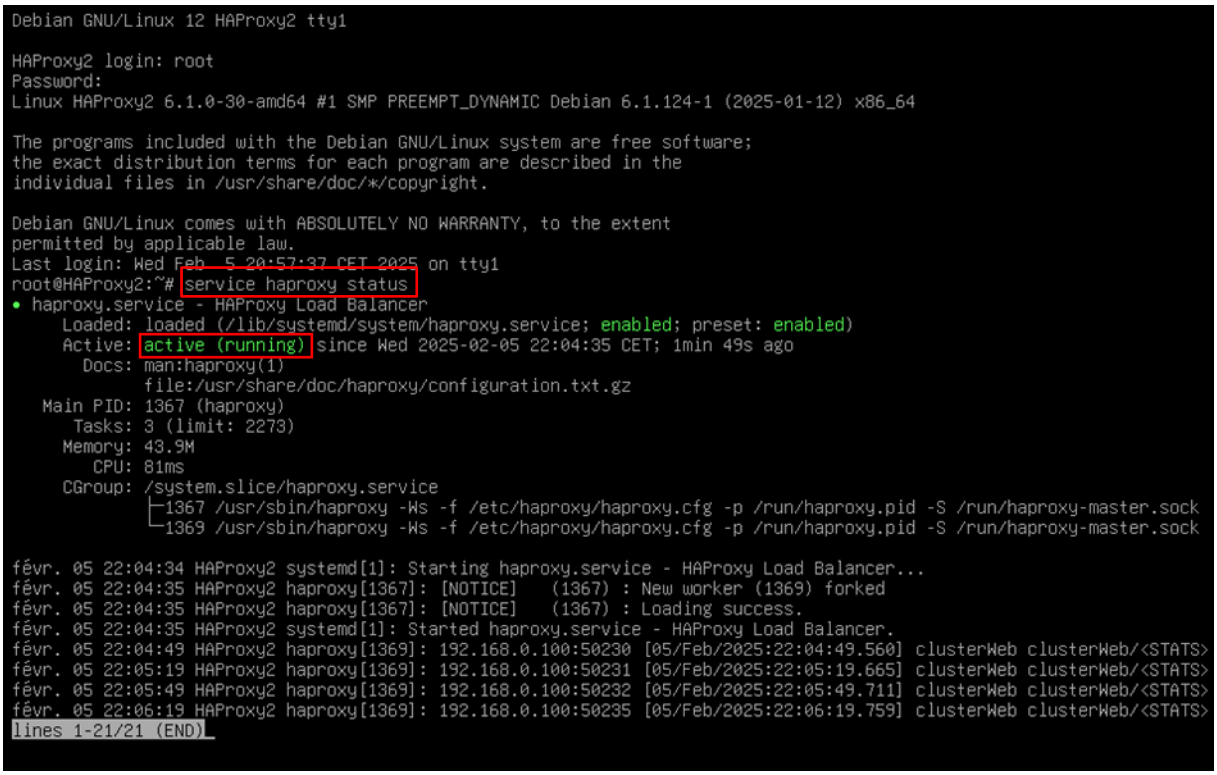
Vérifier que le service HAProxy soit allumé sur HAProxy1 et HAProxy2 en faisant :
Service haproxy status
Il faut qu'il y ait écrit Active running.

HAProxy1



Suite étape 2 :

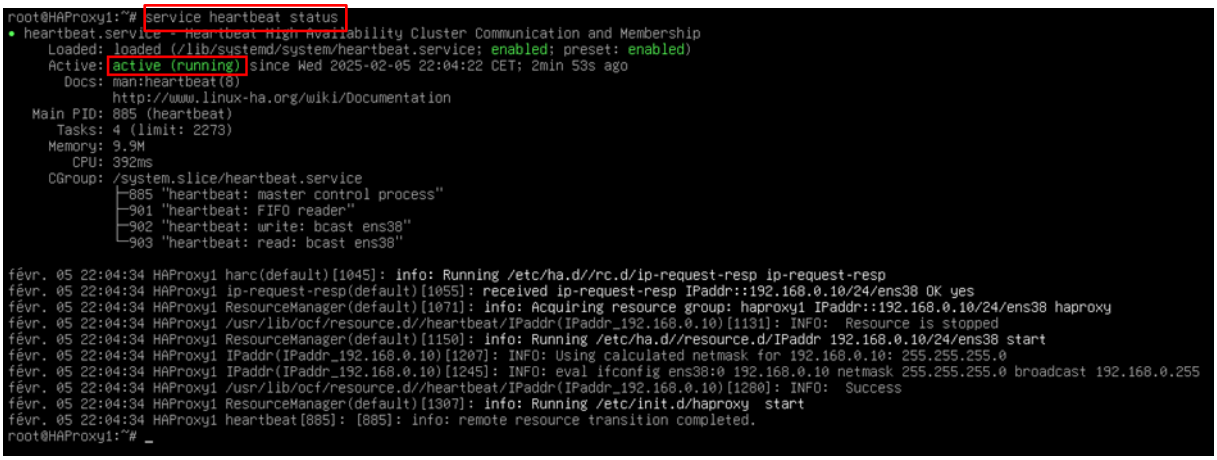
HAProxy2



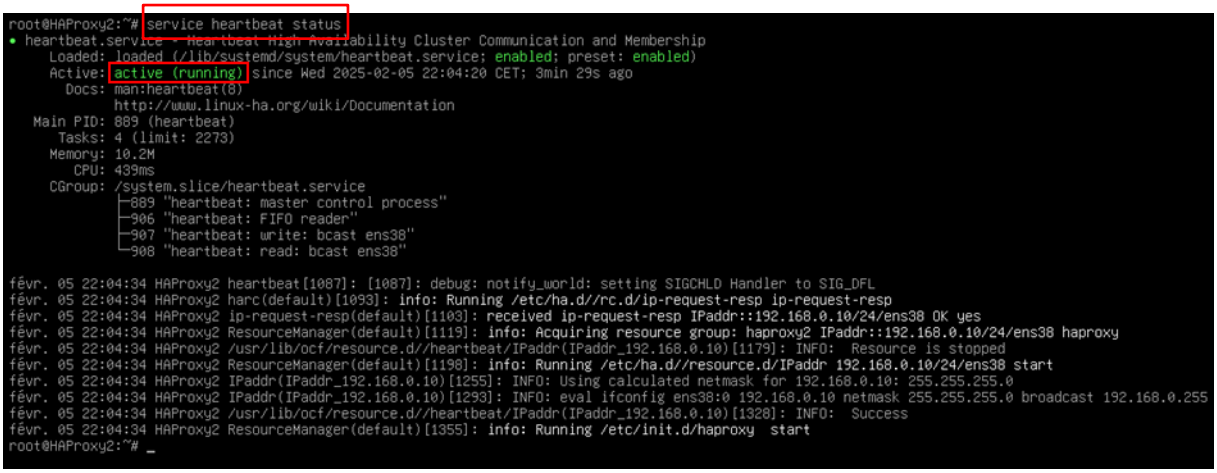
Etape 3 :

Vérifier que le service Heartbeat soit allumé sur HAProxy1 et HAProxy2 en faisant :
Service heartbeat status
Il faut qu'il y ait écrit Active running.

HAProxy 1



HAProxy2



Etape 4 :

Vérifier aussi que les modifications concernant le documentroot d'apache2 des deux machines SRV-WEB dans le fichier nano /etc/apache2/sites-available/000-default.conf ont été enregistrées.

```
root@SRV-WEB1:~# nano /etc/apache2/sites-available/000-default.conf
```

SRV-WEB 1

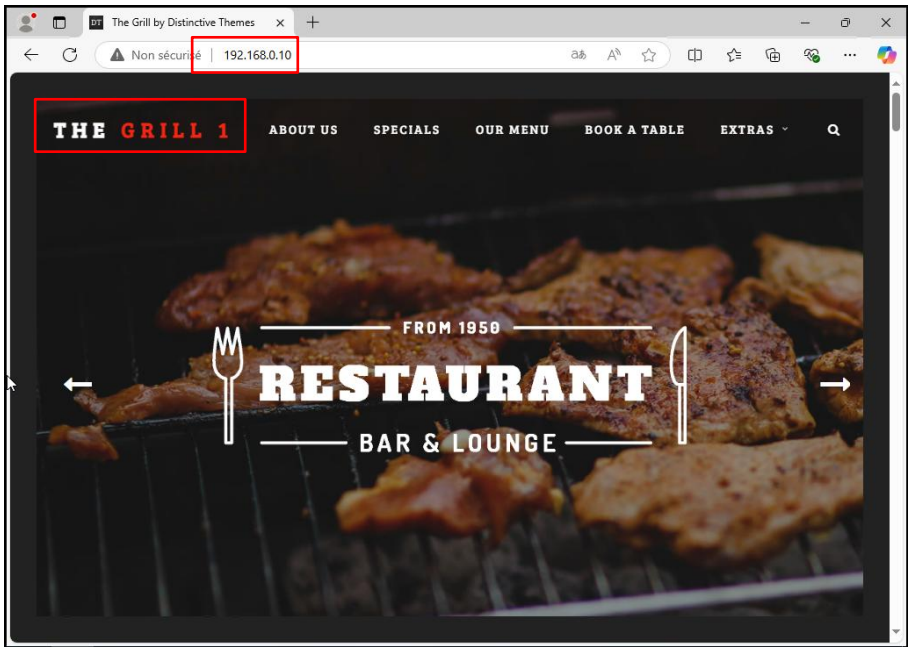
```
 GNU nano 7.2                                  /etc/apache2/sites-available/000-default.conf
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

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

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

SRV-WEB 2

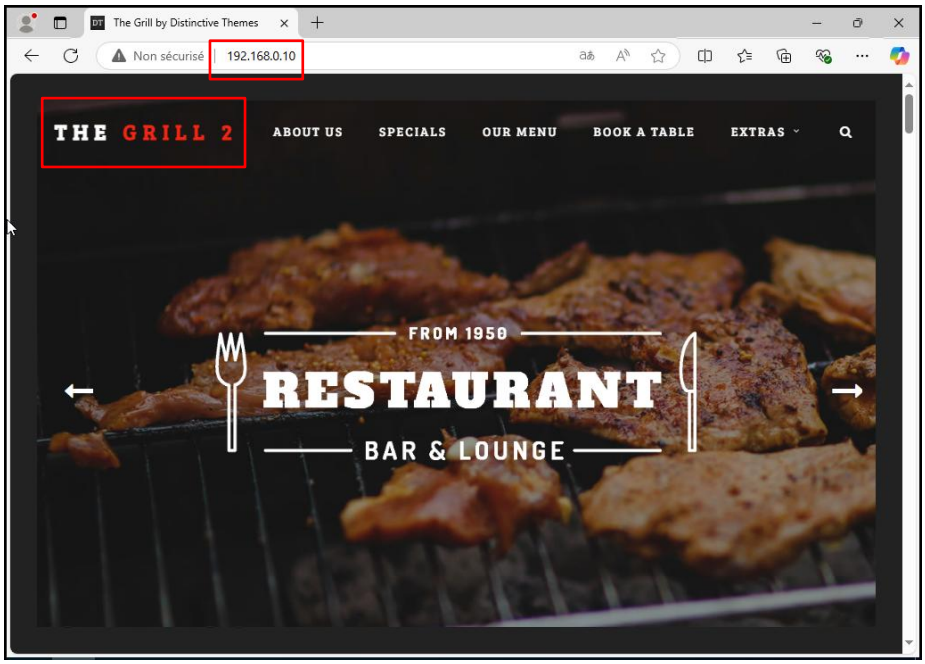
```
 GNU nano 7.2                                  /etc/apache2/sites-available/000-default.conf
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

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

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

Etape 5 :

1. Pour vérifier le balancement, il faut allumer la machine Client Windows, aller dans le navigateur WEB, tapez l'adresse IP Virtuel de Heartbeat (192.168.0.10), le premier site qui s'affiche est « THE GRILL 1 ».



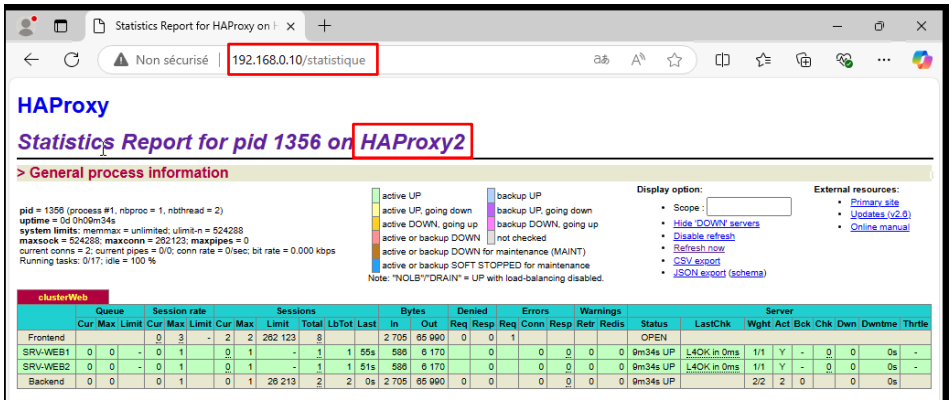
2. Ensuite actualisez la page en tapant la touche « F5 », le deuxième site apparaît « THE GRILL 2 ».



Ainsi, le load balancing est fonctionnel.

Etape 6 :

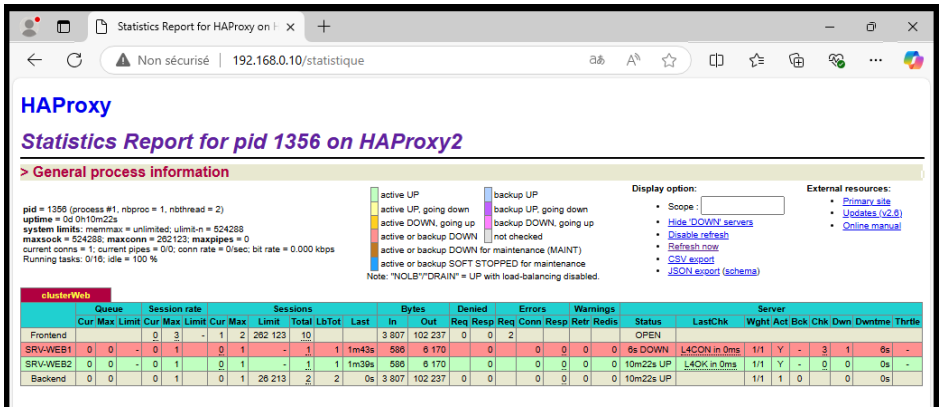
1. On peut également accéder à la page des statistiques à l'aide de l'url @IP/statistique. Le login est « admin:password », une page de statistique avec les deux SRV-WEB1 et 2 sont en vert.



On remarque qu'on se trouve sur HAProxy2

2. Pour faire un test, on peut taper "service apache2 stop" sur l'une des machines.

```
root@SRV-WEB1:~# service apache2 stop
```



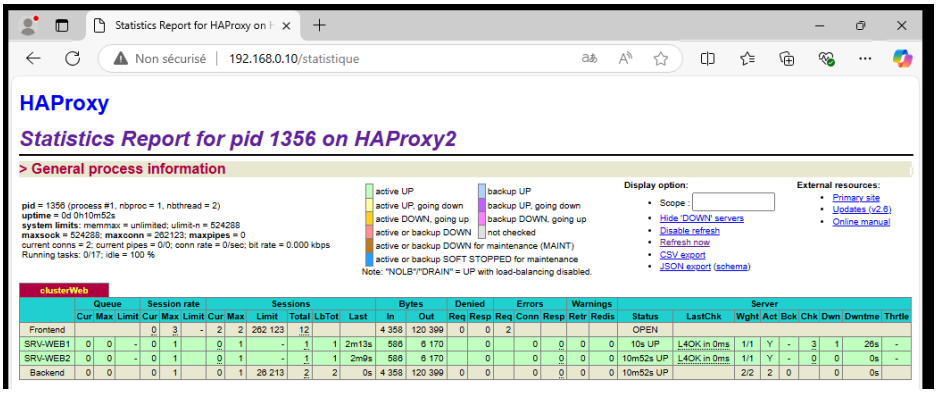
3. Attendre 30 secondes et vérifier que la machine en question passe en état down en rouge dans la page des statistiques.

SRV-WEB1 est en rouge !

Etape 6 (Suite) :

```
root@SRV-WEB1:~# service apache2 start
```

4. On le redémarre en tapant service apache2 start.



5. Attendre 30 secondes et vérifier que la machine en question passe en état running dans la page des statistiques.

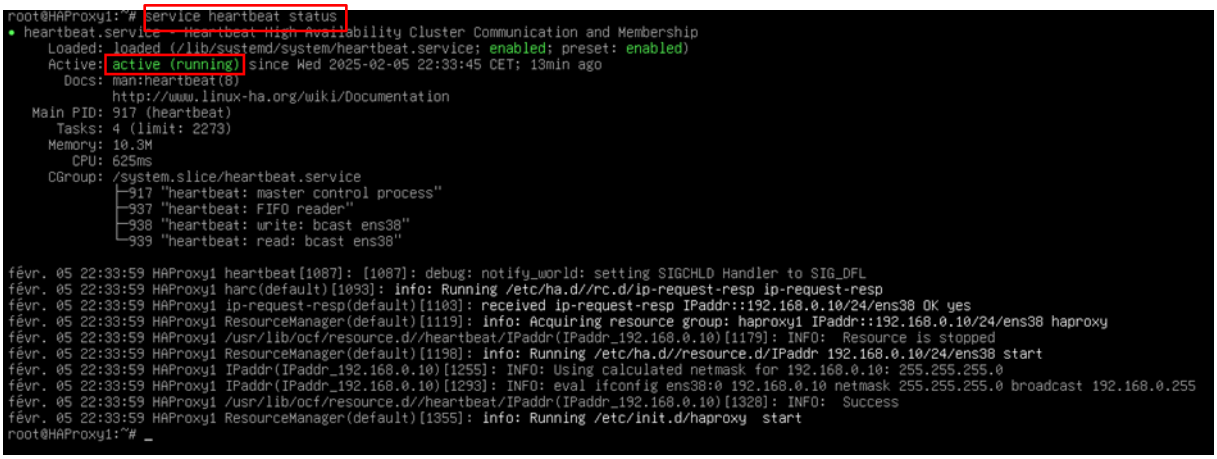
Tout est revenu en vert !

c. *Vérification du fonctionnement du Heartbeat*

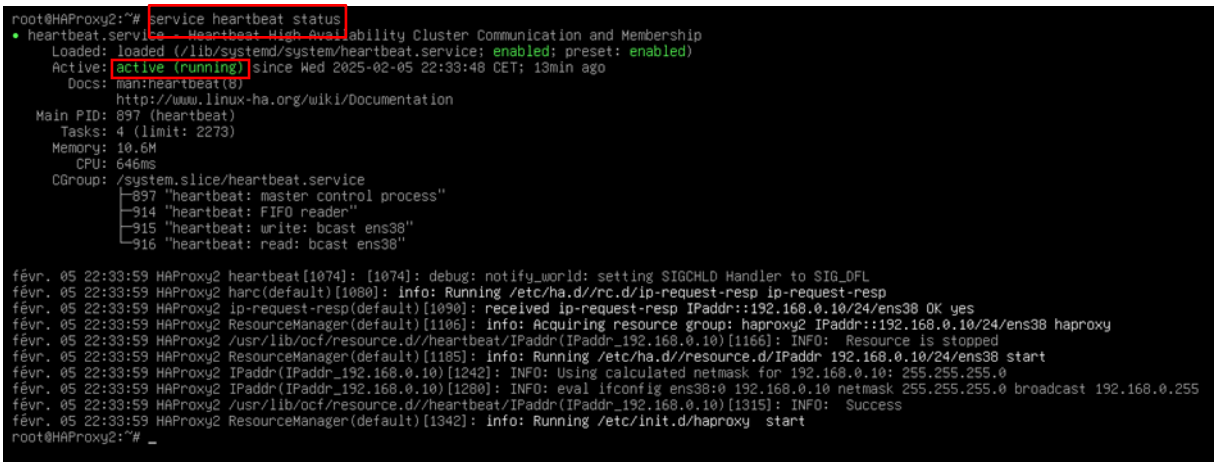
Etape 1 :

Allumez les machines HAProxy1 et 2, on vérifie que le Heartbeat est actif avec « service heartbeat status » et y voir Active running.

HAProxy1

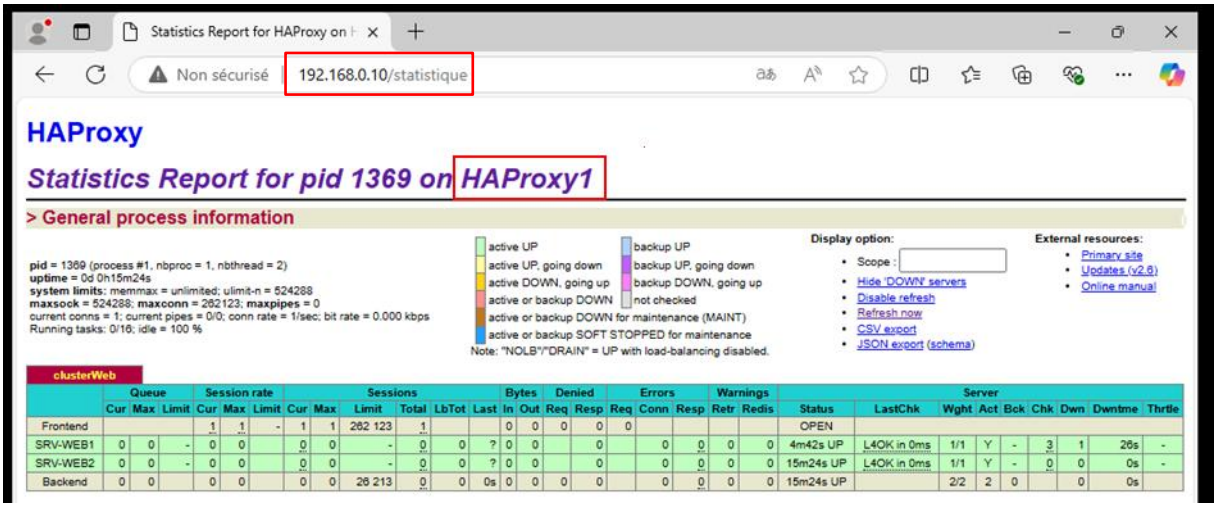


HAProxy2



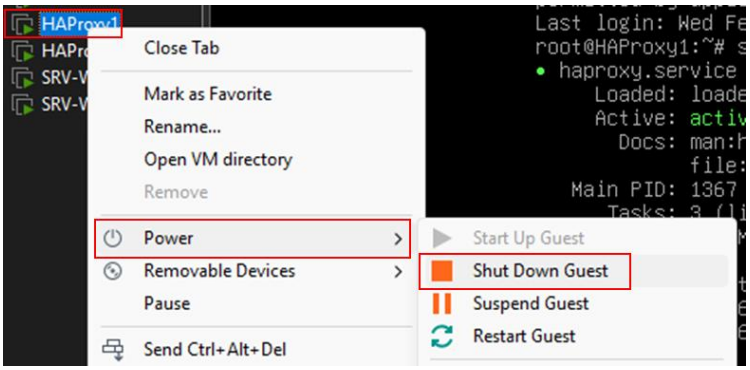
Etape 2 :

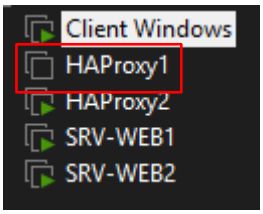
Depuis la machine Client Windows, lancez le navigateur WEB et tapez l'adresse IP Virtuelle : http://192.168.0.10/statistique, on remarque qu'on est sur HAProxy1.



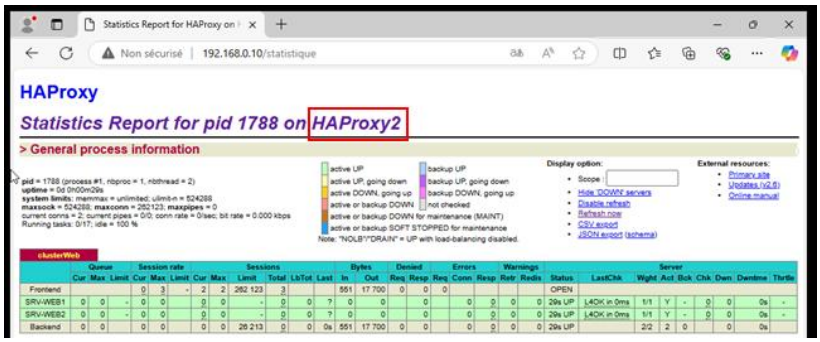
Etape 3 :

1. Pour simuler un crash de HAProxy1, je vais éteindre HAProxy1 attendre un peu et voir si HAProxy2 prend le relais sur le navigateur du Client Windows automatiquement.





2. HAProxy1 est off, et HAProxy2 à pris le relais automatiquement.



Ainsi le load balancing est fonctionnel.

Annexe : Explication détaillant les fonctions des commandes utilisées.

Commande	Fonction	Exemple
Ifup	Démarre une carte réseau	Ifup ens33 allume la carte réseau ens33
Ifdown	Eteins une carte réseau	Ifdown ens33 éteins la carte réseau ens33
Ifdown + ifup	Permet d'éteindre puis rallumer une carte réseau pour lui faire prendre une nouvelle configuration	Ifdown ens33 Ifup ens33 Redémarre la carte réseau
Nano	Modifie un fichier	Nano /etc/hostname permet de modifier le hostname de la machine