

AP – 10 Haproxy – Heartbeat – Master / Slave

Présenté par :

Yassir Chellik ET Gaetan Bracale

Année 2024 - 2025

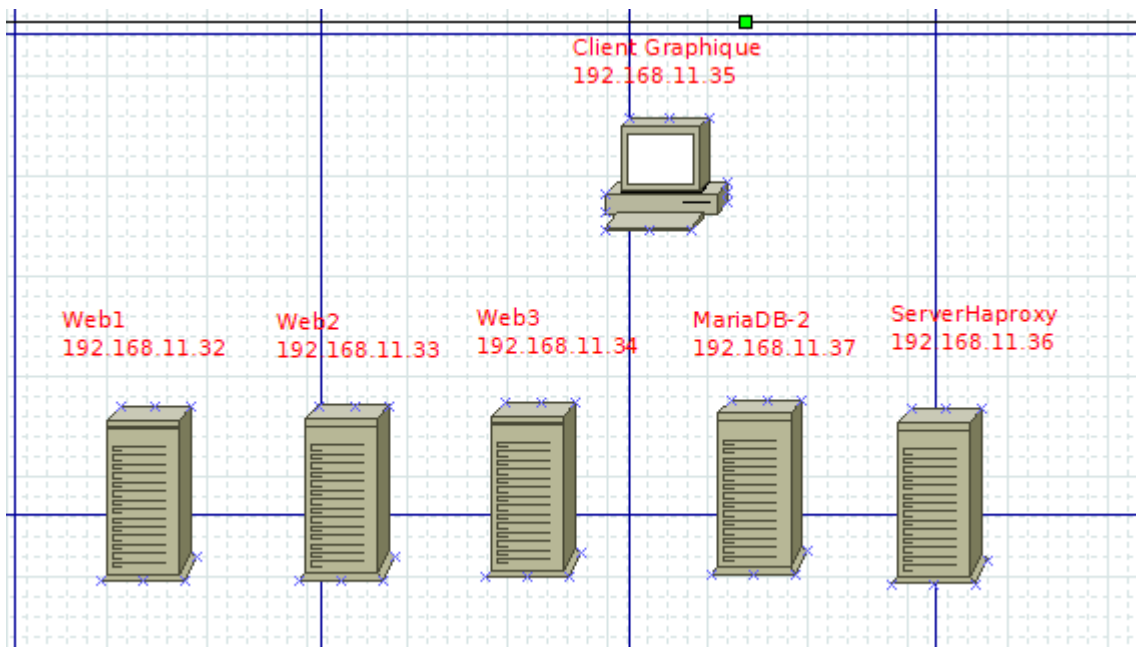
Suite à l'absence de mon camarade j'ai du faire le TP tout seul :

1. Répartir les tâches et créer le Gantt avec tableau infra.

[AP : Yassir Gaetan Gestion VLAN et VMs du contexte "Menuimétal"](#)

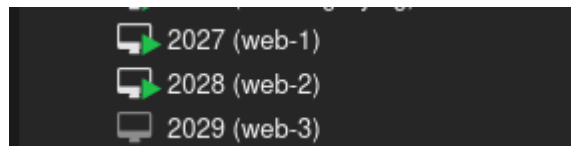
[Ap - 10 Yassir / Gaetan](#)

2. Mettre à jour le schéma réseau.

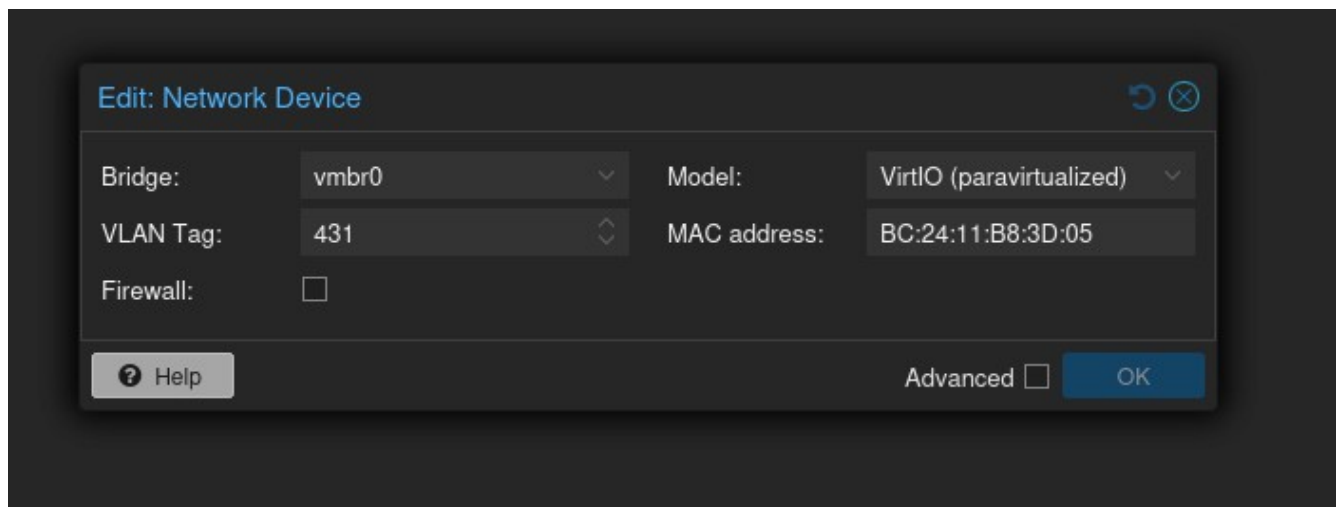


3. Cloner 3 VMs serveurs Web Linux, modifier leur hostname.

Nous avons cloné 3VM linux serveur pour avoir trois serveurs web.



Nos vms seront dans le VLAN LAN (192.168.11.X)



Nous allons vous montrer 1 screen sur 3 pour aller plus vite, j'ai modifié le hostname de ma machine en fonctions du service et du nom du serveur en cohérence avec le projet `/etc/hostname` et j'ai également changé l'interface

réseau de ma machine avec une IP dans le Vlan LAN en
192.168.11.32 dans `/etc/network/interfaces`

Nous avons fait de même avec les autres VM qui seront
en 192.168.11.33 et 192.168.11.34

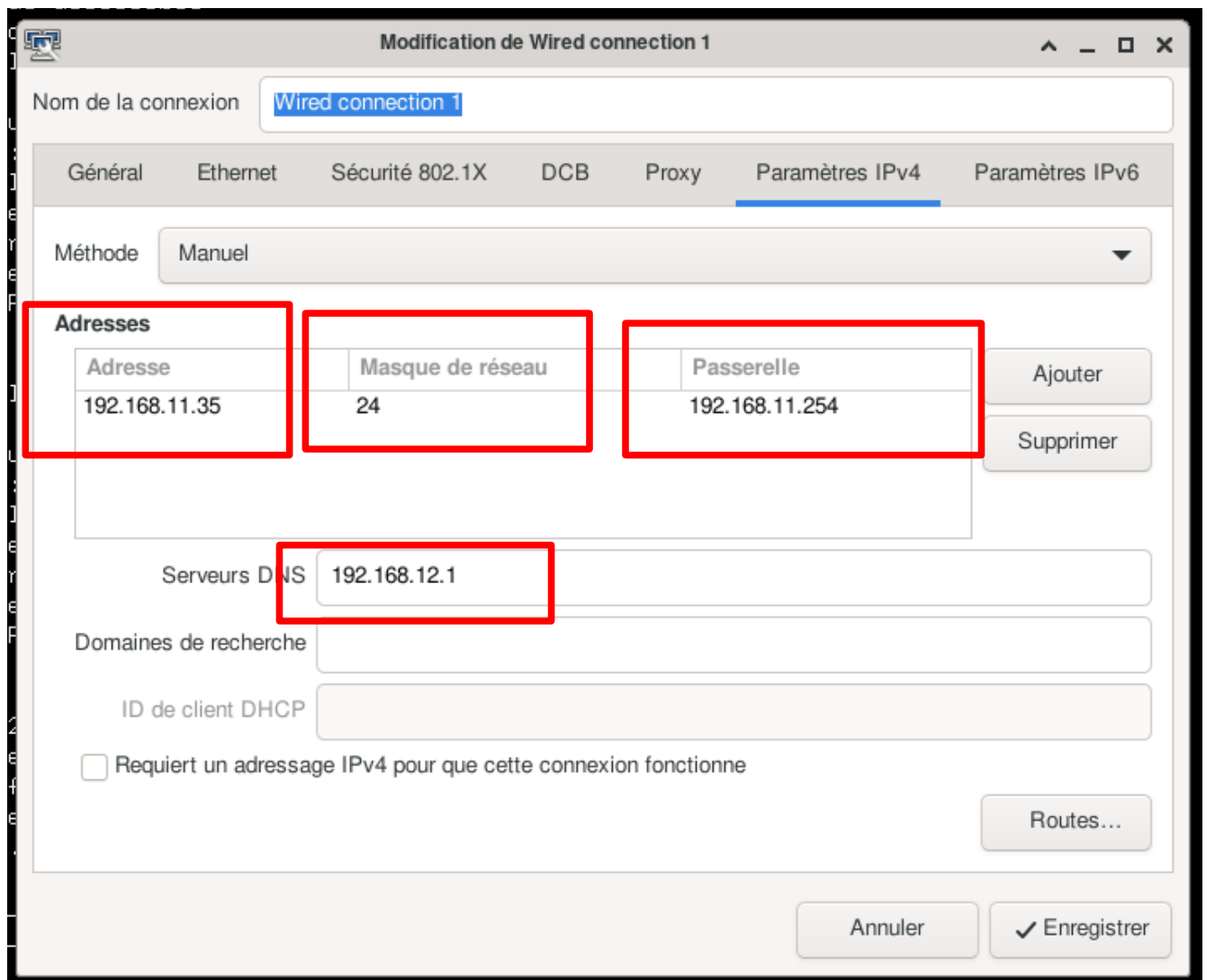
```
sio@web-1:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:b8:3d:05 brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 192.168.11.32/24 brd 192.168.13.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:feb8:3d05/64 scope link
        valid_lft forever preferred_lft forever
sio@web-1:~$ vim /etc/net
netconfig network/ networks
sio@web-1:~$ vim /etc/network/interfaces
sio@web-1:~$ su -
Mot de passe :
root@web-1:~# vim /etc/network/interfaces
root@web-1:~# ping 192.168.12.1
PING 192.168.12.1 (192.168.12.1) 56(84) bytes of data:
64 bytes from 192.168.12.1: icmp_seq=1 ttl=63 time=1.43 ms
^C
--- 192.168.12.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.434/1.434/1.434/0.000 ms
root@web-1:~#
```



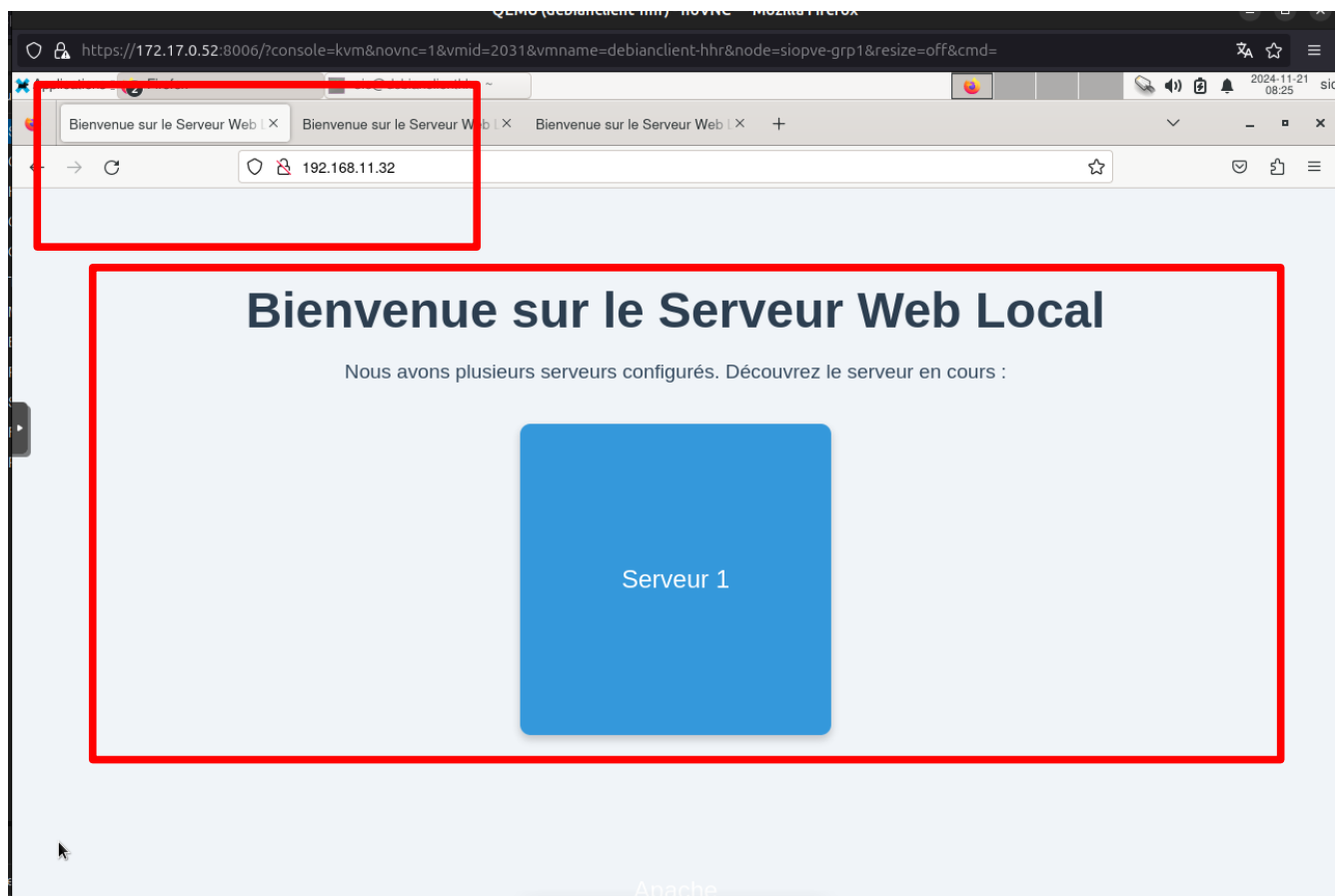
4. Cloner une VM Cliente pour tester les serveurs Web.

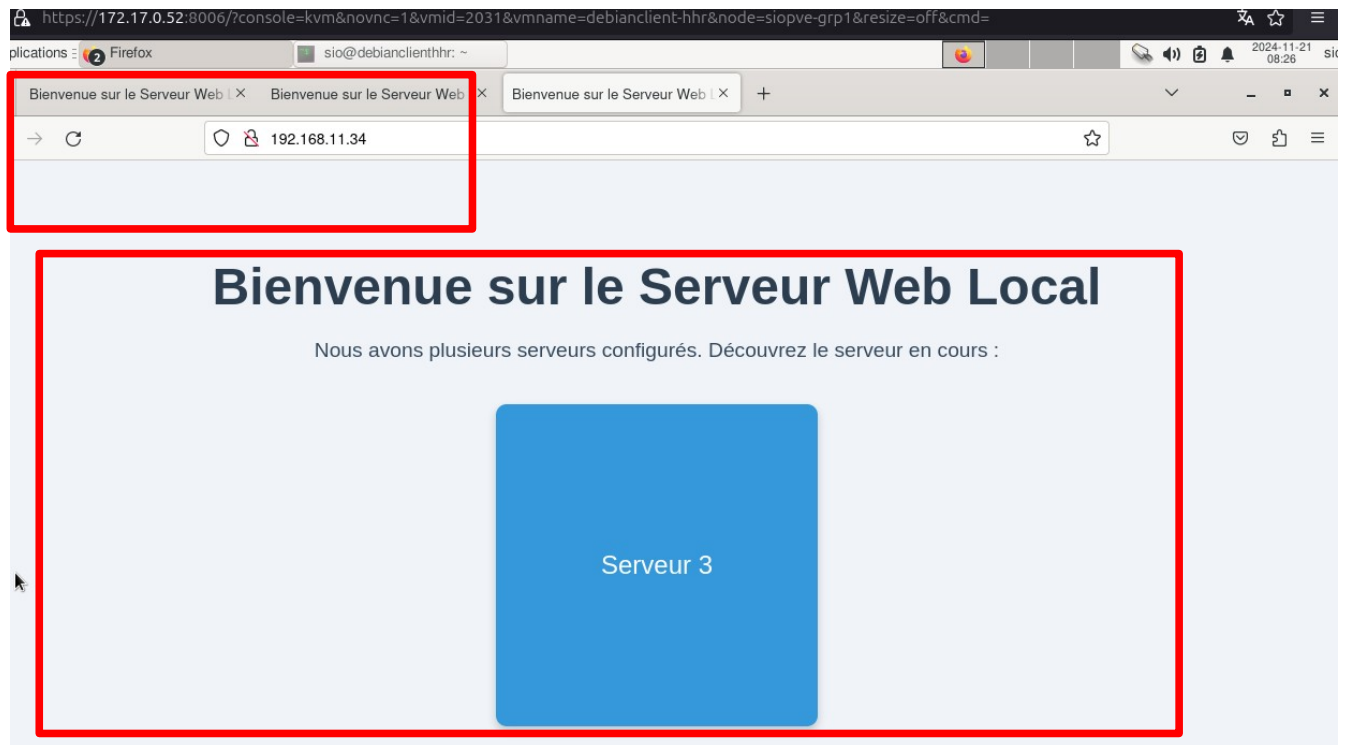
Nous allons cloner une VM cliente XFCE (graphique) pour tester les serveurs web, j'ai modifié le hostname de ma machine en fonctions du contexte et en cohérence avec le projet `/etc/hostname` et j'ai également changé l'interface réseau de ma machine avec une IP dans le Vlan LAN en 192.168.11.35 dans la modification en mode graphique.

```
root@debianclientthhr:~# 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: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:14:df:f6 brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 192.168.11.35/24 brd 192.168.11.255 scope global noprefixroute ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::49b4:8e10:6aa2:dfd6/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
root@debianclientthhr:~# ping 192.168.12.1
PING 192.168.12.1 (192.168.12.1) 56(84) bytes of data.
64 bytes from 192.168.12.1: icmp_seq=1 ttl=63 time=1.35 ms
64 bytes from 192.168.12.1: icmp_seq=2 ttl=63 time=1.24 ms
^C
--- 192.168.12.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 1.239/1.295/1.352/0.056 ms
root@debianclientthhr:~#
```



Depuis notre machine cliente nous avons testé les pages d'accueil des serveurs web en installant **apache2** et en modifiant le fichier HTML dans **/var/www/html** pour préparer une bonne page d'accueil, il a fallu mettre l'adresse ip du serveur web dans le navigateur de ma machine





5. Cloner une VM serveur Haproxy, modifier son hostname et IP.

Nous avons cloner une machine serveur linux pour un serveur Haproxy et nous avons modifié le hostname de

ma machine en fonctions du service et du nom du serveur en cohérence avec le projet `/etc/hostname` et j'ai également changé l'interface réseau de ma machine avec une IP dans le Vlan LAN en 192.168.11.36 dans `/etc/network/interfaces`

```
https://172.17.0.52:8006/?console=kvm&novnc=1&vmid=2033&vmname=ServerHaProxy&node=siopve-grp1&res...
root@serverhaproxy:~# 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: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:c6:ac:8c brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 192.168.11.36/24 brd 192.168.11.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fec6:ac8c/64 scope link
        valid_lft forever preferred_lft forever
root@serverhaproxy:~# _
```

6. Cloner la VM MariaDB, voir les bases de données.

Nous avons également cloner une vm mariaDB pour un second serveur de base de donnée, nous avons changé l'adresse IP en 192.168.11.37 dans le fichier `/etc/network/interfaces` et le hostname dans `/etc/hostname`, nous avons également vérifié que les base de donnée ont bien été cloné sur notre vm en se connectant avec mysql :

- mysql -u root -p
- SHOW DATABASES;

```
QEMU (mariadb-2) - noVNC — Mozilla Firefox
https://172.17.0.52:8006/?console=kvm&novnc=1&vmid=2032&vmname=mariadb-2&node=siopve-grp1&resize=off&c
Server version: 10.3.29-MariaDB-0+deb10u1 Debian 10
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]> show databases
-> Ctrl-C -- exit!
Aborted
root@mariadb-2:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 50
Server version: 10.3.29-MariaDB-0+deb10u1 Debian 10
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| DOKUWIKI |
| glpi     |
| glpidb  |
| information_schema |
| mysql   |
| performance_schema |
| wordpress_db |
+-----+
7 rows in set (0.001 sec)

MariaDB [(none)]> exit
Bye
root@mariadb-2:~# 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
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether bc:24:11:67:b5:44 brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 192.168.11.37/24 brd 192.168.11.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe67:b544/64 scope link
        valid_lft forever preferred_lft forever
root@mariadb-2:~# _
```

7. Référencer les VMs sur le DNS.

Nous avons référencé les vms dans notre serveur dns, et nous avons également testé avec un nslookup et le nom de notre machine référencer, les résultats obtenus sont :
le nom de domaine, l'adresse ip de la machine et du serveur dns

```
@      IN      NS      dns
@      IN      MX 10   servermail
dns    IN      A        192.168.12.1
mysql  IN      A        192.168.11.1
web    IN      A        192.168.12.2
routeur IN     A        192.168.12.254
srvwebwordpress IN  A        192.168.11.3
srvlinuxgpi IN    A        192.168.13.2
servermail IN   A        192.168.12.4
srv-rancid IN   A        192.168.13.7
srv-nagios IN   A        192.168.13.8
omvdebian IN   A        192.168.11.28
serverdhcpdebian IN A        192.168.11.29
serverradius IN  A        192.168.13.9
serv-suslog IN   A        192.168.11.31
web-1  IN      A        192.168.11.32
web-2  IN      A        192.168.11.33
web-3  IN      A        192.168.11.34
mariadb-2 IN   A        192.168.11.37

smtp   IN      CNAME   servermail
imap   IN      CNAME   servermail
~
~
~
~
~
~
~
"/var/cache/bind/db.menuimetal.fr" 41L, 748B écrit(s)
root@dns:~# systemctl restart bind9
root@dns:~# nslookup
> web-1
Server:          192.168.12.1
Address:         192.168.12.1#53

Name:   web-1.menuimetal.fr
Address: 192.168.11.32
>
```

Nous avons également fait un test de communication entre notre machine serveur web et le serveur dns avec un ping 192.168.12.1

```
root@web-1:~# ping 192.168.12.1
PING 192.168.12.1 (192.168.12.1) 56(84) bytes of data.
64 bytes from 192.168.12.1: icmp_seq=1 ttl=63 time=0.910 ms
^C
--- 192.168.12.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.910/0.910/0.910/0.000 ms
root@web-1:~#
```

8. Ajouter les VMs au serveur GLPI et nagios.

Afin d'installer l'agent glpi, il faut référencer le proxy du lycée pour permettre d'installer le paquet : `/etc/wgerc` ensuite il faudra télécharger du Script d'Installation de l'Agent GLPI :

Téléchargez le script d'installation de l'agent GLPI à l'aide de la commande `wget`. Ce script en Perl permet de configurer et d'installer l'agent GLPI sur votre machine Debian :

`wget`
<https://github.com/glpi-project/glpi-agent/releases/download/1.11/glpi-agent-1.11-linux-installer.pl>

Installez de Perl :

Le script d'installation étant écrit en Perl (d'où l'extension .pl), assurez-vous que Perl est installé sur votre machine. Si ce n'est pas le cas, vous pouvez l'installer avec la commande suivante :

```
apt install perl
```

Exécutez du Script d'Installation (avec perl) :

Lancez le script Perl pour installer l'agent GLPI. Ce script vous guidera tout au long du processus d'installation, en configurant l'agent pour qu'il envoie les informations à votre serveur GLPI.

```
perl glpi-agent-1.7.1-linux-installer.pl -s  
http://192.168.13.2/ --runnow --install
```

```

40 wget https://github.com/glpi-project/glpi-agent/releases/download/1.7.1/glpi-agent-1.7.1-linux-installer.pl
41 ls
42 perl
43 perl glpi-agent-1.7.1-linux-installer.pl
44 perl glpi-agent-1.7.1-linux-installer.pl -s http://192.168.13.2/glpi --runnow --install
45 exit

```

Ensuite redémarrer le service glpi agent :

systemctl restart glpi-agent

```

root@debianclienthhr:~# systemctl status glpi-agent.service
● glpi-agent.service - GLPI agent
   Loaded: loaded (/lib/systemd/system/glpi-agent.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-11-19 16:29:42 CET; 1 day 16h ago
     Docs: man:glpi-agent
  Main PID: 501 (glpi-agent: wai)
    Tasks: 1 (limit: 2285)
   Memory: 96.3M
      CPU: 35.193s
   CGroup: /system.slice/glpi-agent.service
           └─501 "glpi-agent: waiting"

nov. 19 17:15:40 debianclienthhr glpi-agent[1828]: Use of uninitialized value $hostname in substitution (s///) at /usr/share/glpi-agent/lib/GLPI/Agent/Tools/Hostname.pm line 64.
nov. 19 17:15:40 debianclienthhr glpi-agent[1828]: Use of uninitialized value $hostname in index at /usr/share/glpi-agent/lib/GLPI/Agent/Task/Inventory/Generic/Domain.pm line 50.
nov. 19 17:15:54 debianclienthhr glpi-agent[501]: [info] target server0: next run: Wed Nov 20 16:21:46 2024 - http://192.168.13.2/glpi
nov. 20 16:21:46 debianclienthhr glpi-agent[501]: [info] target server0: server http://192.168.13.2/glpi
nov. 20 16:21:46 debianclienthhr glpi-agent[501]: [info] sending contact request to server0
nov. 20 16:21:47 debianclienthhr glpi-agent[3605]: [info] running task Inventory
nov. 20 16:21:47 debianclienthhr glpi-agent[3605]: [info] New inventory from debian-2024-11-08-15-03-29 for server0
nov. 20 16:21:47 debianclienthhr glpi-agent[3605]: Use of uninitialized value $hostname in substitution (s///) at /usr/share/glpi-agent/lib/GLPI/Agent/Tools/Hostname.pm line 64.
nov. 20 16:21:47 debianclienthhr glpi-agent[3605]: Use of uninitialized value $hostname in index at /usr/share/glpi-agent/lib/GLPI/Agent/Task/Inventory/Generic/Domain.pm line 50.
nov. 20 16:22:00 debianclienthhr glpi-agent[501]: [info] target server0: next run: Thu Nov 21 16:14:29 2024 - http://192.168.13.2/glpi
root@debianclienthhr:~#

```

Enfin voici nos machines monté sur glpi :

<input type="checkbox"/>	web-1	Entité racine	QEMU	QEMU	Standard PC (i440FX + PIIX, 1996)
<input type="checkbox"/>	web-2	Entité racine	QEMU	QEMU	Standard PC (i440FX + PIIX, 1996)
<input type="checkbox"/>	web-3	Entité racine	QEMU	QEMU	Standard PC (i440FX + PIIX, 1996)

<input type="checkbox"/>	debianclientthr	Entité racine
<input type="checkbox"/>	DESKTOP-T2R477L	Entité racine
<input type="checkbox"/>	dns	Entité racine
<input type="checkbox"/>	DOKUWIKI	Entité racine
<input type="checkbox"/>	mariadb	Entité racine
<input type="checkbox"/>	mariadb-2	Entité racine
<input type="checkbox"/>	omvdebian	Entité racine
<input type="checkbox"/>	routeur	Entité racine
<input type="checkbox"/>	serverdhcpdebian	Entité racine
<input type="checkbox"/>	serverhaproxy	Entité racine

```

root@web-1:/usr/lib/nagios/plugins# systemctl status nagios-nrpe-server
● nagios-nrpe-server.service - Nagios Remote Plugin Executor
   Loaded: loaded (/lib/systemd/system/nagios-nrpe-server.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-11-21 09:00:53 CET; 8s ago
     Docs: http://www.nagios.org/documentation
   Main PID: 9300 (nrpe)
    Tasks: 1 (limit: 2306)
   Memory: 1.2M
      CPU: 15ms
   CGroup: /system.slice/nagios-nrpe-server.service
           └─9300 /usr/sbin/nrpe -c /etc/nagios/nrpe.cfg -f

nov. 21 09:00:53 web-1 systemd[1]: Started nagios-nrpe-server.service - Nagios Remote Plugin Executor.
nov. 21 09:00:53 web-1 nrpe[9300]: Starting up daemon
nov. 21 09:00:53 web-1 nrpe[9300]: Server listening on 0.0.0.0 port 5666.
nov. 21 09:00:53 web-1 nrpe[9300]: Server listening on :: port 5666.
nov. 21 09:00:53 web-1 nrpe[9300]: Listening for connections on port 5666
nov. 21 09:00:53 web-1 nrpe[9300]: Allowing connections from: 127.0.0.1,192.168.13.8
root@web-1:/usr/lib/nagios/plugins# █

```

9. Accéder aux VMs via SSH.

Les images déjà mentionnés dans le compte rendu nous prouve déjà que le SSH est accessible si vous avez un problèmes avec l'accès du ssh il faudra :

vérifier que le service SSH est actif :

```
sudo systemctl status ssh
```

Si le service est arrêté, le démarrer :

```
sudo systemctl start ssh
```

Analyse des logs SSH :

Consulter les logs du service SSH pour identifier d'éventuelles erreurs :

```
sudo journalctl -u ssh
```

Validation du fichier de configuration SSH :

Vérifier le fichier `/etc/ssh/sshd_config` pour des erreurs ou des modifications suspectes.
Recharger le service si des modifications sont apportées :

```
sudo systemctl restart ssh
```

Tester la connectivité réseau :

Vérifier si la machine est atteignable via ping :

```
ping <adresse_ip>
```

Tester la connexion SSH directement :

```
ssh -v <user>@<adresse_ip>
```

10. Sauvegarder les bases de données du serveur maître avec mysqldump.

Sur le serveur maître

a. Sauvegarder toutes les bases de données

Connectez-vous au serveur maître via SSH. Pour rappel le serveur maître est le serveur mariadb de base donc le premier en 192.168.11.1

Exécutez la commande suivante pour sauvegarder toutes les bases :

```
mysqldump -u root -p --all-databases > sauvegarde.sql
```

- **--all-databases** : sauvegarde toutes les bases présentes sur le serveur.
- **> sauvegarde.sql** : enregistre la sortie dans un fichier nommé **sauvegarde.sql**.

Si nos bases contiennent des données, elles seront écrites dans ce fichier.

```
root@mariadb:~# mysqldump -u root -p --all-databases > sauvegarde.sql
```

Voici un exemple du contenu du fichier de sauvegarde créé il y'a les base de donnée et les éléments comme les tables ou utilisateurs crée :

```
--
-- Current Database: `DOKUWIKI`
--
CREATE DATABASE /*!32312 IF NOT EXISTS*/ `DOKUWIKI` /*!40100 DEFAULT CHARACTER SET utf8mb4 */;
USE `DOKUWIKI`;
--
-- Table structure for table `utilisateurs`
--
DROP TABLE IF EXISTS `utilisateurs`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `utilisateurs` (
  `username` varchar(50) NOT NULL,
  `password` varchar(255) NOT NULL,
  `name` varchar(100) DEFAULT NULL,
  `email` varchar(100) DEFAULT NULL,
  `groups` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`username`)
)

```

Read 1702

```
LOCK TABLES `utilisateurs` WRITE;
/*!40000 ALTER TABLE `utilisateurs` DISABLE KEYS */;
INSERT INTO `utilisateurs` VALUES ('admin','933d2f2d1f1474f093f574aee60352d7',NULL,'admin@example.com','admin'),('testuser','motdepassecrypt','Test User','testuser@example.com','users');
/*!40000 ALTER TABLE `utilisateurs` ENABLE KEYS */;
UNLOCK TABLES;

```

11. Configurer le serveur maître pour la réplication

Éditez le fichier de configuration de MySQL/MariaDB :

```
nano /etc/mysql/my.cnf
```

Ajoutez ou modifiez les lignes suivantes dans la section
[mysqld] :

```
server-id=1 log_bin=/var/log/mysql/mysql-
```

bin.log

server-id=1 : identifiant unique pour ce serveur.
Chaque serveur doit avoir un **server-id** différent pour éviter les conflits dans la réplication.

log_bin : active les journaux binaires utilisés pour la réplication. Ces journaux sont nécessaires pour que le serveur maître puisse transmettre les modifications aux serveurs slaves dans une configuration de réplication

```
[mysqld]
server-id=1 #Identifie de manière unique le maitre
log_bin=/var/log/mysql/mysql-bin.log # active les journaux né cessaires pour la répliquati
```

Il faut ensuite redémarrer le service mysql pour les changements :

```
mariadb.service - MariaDB 10.3.29 database server
Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
Active: active (running) since Thu 2024-11-21 09:44:36 CET; 7s ago
Docs: man:mysql(8)
      https://mariadb.com/kb/en/library/systemd/
Process: 13119 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/run/mysql (code=exited, status=0/SUCCESS)
Process: 13120 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited, status=0/SUCCESS)
Process: 13122 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && VAR= || VAR='cd /usr/bin/..; /usr/bin/galera_recovery'; [ $? -eq 0 ] && systemctl set-environment _WSREP
Process: 13211 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited, status=0/SUCCESS)
Process: 13213 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/SUCCESS)
Main PID: 13179 (mysqld)
Status: "Taking your SQL requests now..."
Tasks: 32 (limit: 2322)
Memory: 106.2M
CPU: 623ms
CGroup: /system.slice/mariadb.service
        └─13179 /usr/sbin/mysqld

Nov 21 09:44:36 mariadb systemd[1]: Starting MariaDB 10.3.29 database server...
Nov 21 09:44:36 mariadb mysqld[13179]: 2024-11-21 9:44:36 0 [Note] /usr/sbin/mysqld (mysqld 10.3.29-MariaDB-0+deb10u1-log) starting as process 13179 ...
Nov 21 09:44:36 mariadb systemd[1]: Started MariaDB 10.3.29 database server.
Nov 21 09:44:36 mariadb /etc/mysql/debian-start[13215]: Upgrading MySQL tables if necessary.
Nov 21 09:44:36 mariadb /etc/mysql/debian-start[13218]: /usr/bin/mysql_upgrade: the '--basedir' option is always ignored
Nov 21 09:44:36 mariadb /etc/mysql/debian-start[13218]: Looking for 'mysql' as: /usr/bin/mysql
Nov 21 09:44:36 mariadb /etc/mysql/debian-start[13218]: Looking for 'mysqlcheck' as: /usr/bin/mysqlcheck
Nov 21 09:44:36 mariadb /etc/mysql/debian-start[13218]: This installation of MySQL is already upgraded to 10.3.29-MariaDB, use --force if you still need to run mysql_upgrade
Nov 21 09:44:36 mariadb debian-start[13232]: WARNING: tempfile is deprecated; consider using mktmp instead.
root@mariadb:/etc/mysql# mysql -u root -p
```

```
/usr/bin/mysql_upgrade: the '--basedir' option is always ignored
Looking for 'mysql' as: /usr/bin/mysql
Looking for 'mysqlcheck' as: /usr/bin/mysqlcheck
This installation of MySQL is already upgraded to 10.3.29-MariaDB, use --force if you still need to run mysql_upgrade
```

Ensuite il faut crée un utilisateur dédié à la réplication :
Connectez-vous à MySQL :

```
mysql -u root -p
```

Ensuite, exécutez :

```
CREATE USER 'replicateur'@'%' IDENTIFIED BY 'mot_de_passe';
```

```
GRANT REPLICATION SLAVE ON *.* TO 'replicateur'@'%';
```

FLUSH PRIVILEGES;

replicateur : nom de l'utilisateur pour la réplication.

mot_de_passe : mot de passe pour cet utilisateur j'aurai pue choisir sio.

% : permet à cet utilisateur de se connecter depuis n'importe quelle IP.

```
root@mariadb:/etc/mysql# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 45
Server version: 10.3.29-MariaDB-0+deb10u1-log Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE USER 'replicateur'@'%' IDENTIFIED BY 'mot_de_passe';
Query OK, 0 rows affected (0.022 sec)

MariaDB [(none)]> GRANT REPLICATION SLAVE ON *.* TO 'replicateur'@'%';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> □
```

Obtenez l'état actuel du maître pour configurer le slave :

SHOW MASTER STATUS;

File : Nom du fichier binaire de journalisation (binary log) en cours. Ce fichier contient les transactions enregistrées sur le maître, qui seront envoyées aux esclaves.

Position : Position actuelle dans le fichier binaire. Elle indique où la réplication doit reprendre pour synchroniser un esclave.

Binlog_Do_DB : Liste des bases de données incluses dans la réplication (si configuré).

Binlog_Ignore_DB : Liste des bases de données exclues de la réplication.

```
MariaDB [(none)]> SHOW MASTER STATUS;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000001 |      328 |              |                  |
+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

12. Copier la sauvegarde sur le serveur slave.

Sur le serveur slave (mariadb-2)

Copier la sauvegarde depuis le maître

Transférez le fichier sauvegarde.sql sur le serveur slave avec scp :

```
scp sauvegarde.sql  
sio21@192.168.11.37:/tmp/
```

scp : commande pour copier un fichier via SSH.

sio21 : utilisateur SSH du serveur slave.

192.168.11.37 : adresse IP ou nom du serveur slave.

/tmp/ : chemin où stocker le fichier sur le slave.

ATTENTION !

Dans le chemin il ne faut pas spécifier le chemin /etc/mysql car vous n'aurez pas les droits même en root, il faut d'abord le mettre sur un autre répertoire et aller sur notre machine slave et le déplacer, ne fais pas la même erreur que moi !

```
root@mariadb:~# scp sauvegarde.sql sio21@192.168.11.37:/tmp/  
sio21@192.168.11.37's password:  
sauvegarde.sql 100% 3089KB 108.0MB/s 00:00  
root@mariadb:~#  
root@mariadb:~# █
```

Vérifier le fichier de sauvegarde sur le serveur slave :

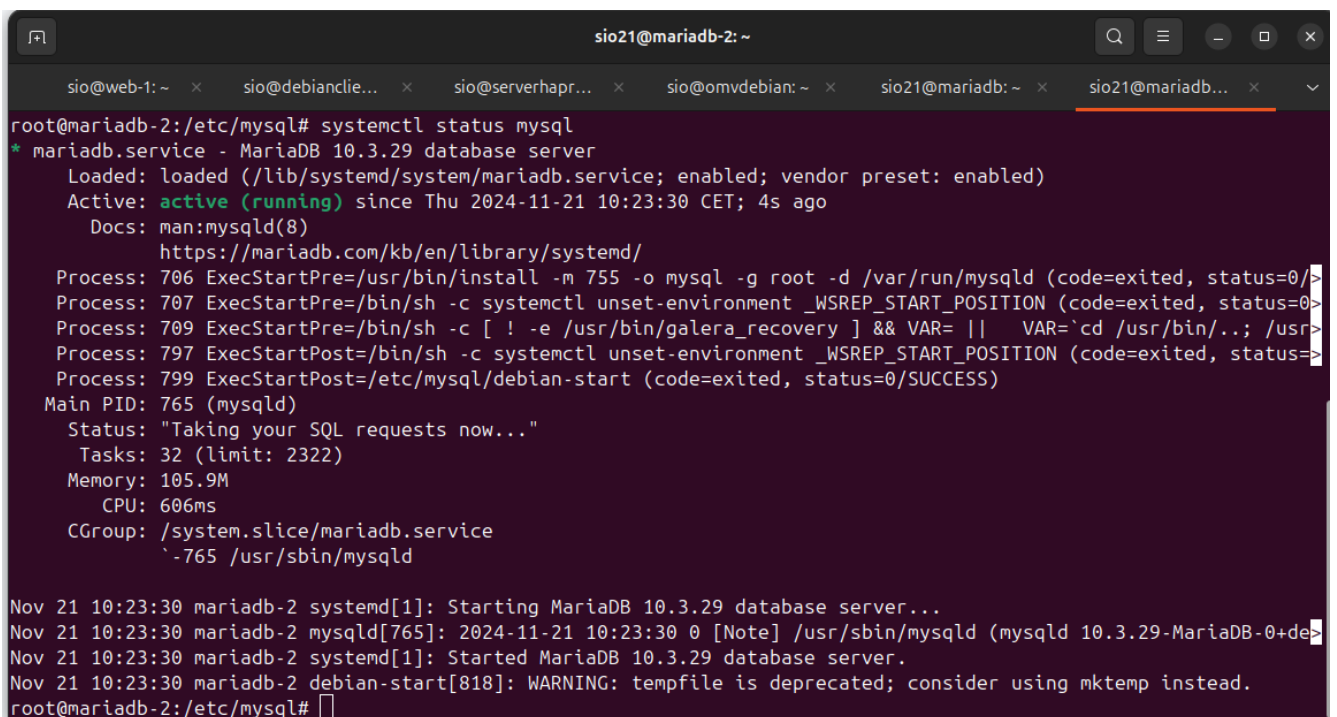
```
root@mariadb-2:~# cd /tmp
root@mariadb-2:/tmp# ls
sauvegarde.sql
systemd-private-1f3dc1596f0e4dcfbca0bf0a6ef7142c-systemd-logind.service-XZuZTf
systemd-private-1f3dc1596f0e4dcfbca0bf0a6ef7142c-systemd-timesyncd.service-mUMamj
root@mariadb-2:/tmp# cp sauvegarde.sql /etc/mysql/
root@mariadb-2:/tmp# ls /etc/mysql/
conf.d  debian-start  debian.cnf  mariadb.cnf  mariadb.conf.d  my.cnf  my.cnf.fallback  sauvegarde.sql
root@mariadb-2:/tmp#
```

13. Configurer le serveur slave pour la réplication.

Nous avons fait de même pour le serveur slave mais lui avons attribué un ID différent du premier pour avoir une bonne réplication et éviter que le serveur maître confonde un esclave avec lui-même, créant des boucles ou des erreurs dans le flux de réplication.

```
[mysqld]
server-id=2
log_bin=/var/log/mysql/mysql-bin.log
```

Il faut ensuite voir le status mysql, pour voir si il n'y a pas d'erreur ou autre : **systemctl restart/status mysql**

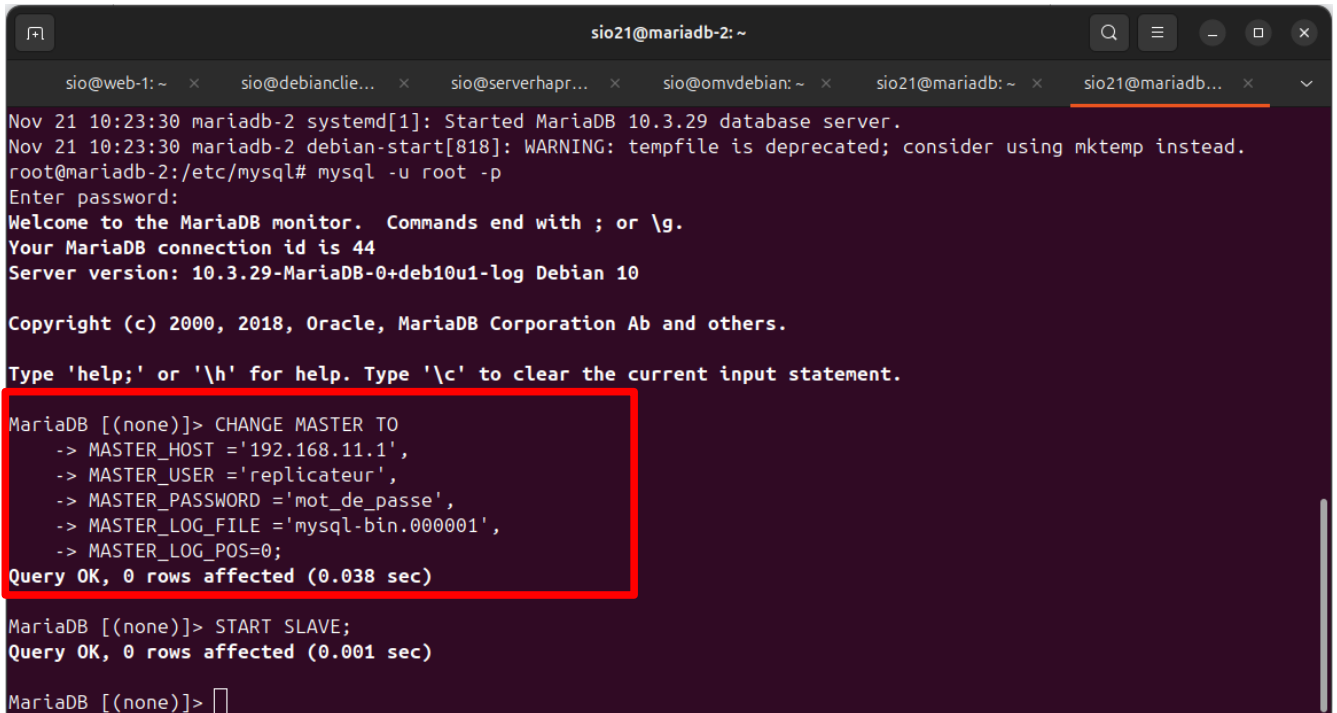


```
sio21@mariadb-2: ~
sio@web-1: ~ x sio@debianclie... x sio@serverhapr... x sio@omvdebian: ~ x sio21@mariadb: ~ x sio21@mariadb... x
root@mariadb-2:/etc/mysql# systemctl status mysql
* mariadb.service - MariaDB 10.3.29 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2024-11-21 10:23:30 CET; 4s ago
     Docs: man:mysqld(8)
           https://mariadb.com/kb/en/library/systemd/
   Process: 706 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/run/mysql (code=exited, status=0/>>
   Process: 707 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited, status=0/>>
   Process: 709 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && VAR=|| VAR='cd /usr/bin/..; /usr>
   Process: 797 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_START_POSITION (code=exited, status=0/>>
   Process: 799 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/SUCCESS)
 Main PID: 765 (mysqld)
    Status: "Taking your SQL requests now..."
     Tasks: 32 (limit: 2322)
  Memory: 105.9M
     CPU: 606ms
    CGroup: /system.slice/mariadb.service
            └─765 /usr/sbin/mysqld

Nov 21 10:23:30 mariadb-2 systemd[1]: Starting MariaDB 10.3.29 database server...
Nov 21 10:23:30 mariadb-2 mysqld[765]: 2024-11-21 10:23:30 0 [Note] /usr/sbin/mysqld (mysqld 10.3.29-MariaDB-0+de>
Nov 21 10:23:30 mariadb-2 systemd[1]: Started MariaDB 10.3.29 database server.
Nov 21 10:23:30 mariadb-2 debian-start[818]: WARNING: tempfile is deprecated; consider using mktemp instead.
root@mariadb-2:/etc/mysql#
```

Il faut ensuite configurez le slave avec les informations du maître, par exemple pour une connectivité et une

réplication il faut préciser l'adresse ip du maitre, le nom d'utilisateur crée, le mot de passe, le fichier de log sql et la position qui doit être similaire quand ont à fais le SHOW MASTER STATUS.



```
sio21@mariadb-2: ~
Nov 21 10:23:30 mariadb-2 systemd[1]: Started MariaDB 10.3.29 database server.
Nov 21 10:23:30 mariadb-2 debian-start[818]: WARNING: tempfile is deprecated; consider using mktemp instead.
root@mariadb-2:/etc/mysql# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 44
Server version: 10.3.29-MariaDB-0+deb10u1-log Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CHANGE MASTER TO
-> MASTER_HOST = '192.168.11.1',
-> MASTER_USER = 'replicateur',
-> MASTER_PASSWORD = 'mot_de_passe',
-> MASTER_LOG_FILE = 'mysql-bin.000001',
-> MASTER_LOG_POS=0;
Query OK, 0 rows affected (0.038 sec)

MariaDB [(none)]> START SLAVE;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> 
```

Nous pouvons voir que depuis notre slave ont peux apercevoir le status de notre maître :

```
sio21@mariadb-2: ~
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CHANGE MASTER TO
-> MASTER_HOST = '192.168.11.1',
-> MASTER_USER = 'replicateur',
-> MASTER_PASSWORD = 'mot_de_passe',
-> MASTER_LOG_FILE = 'mysql-bin.000001',
-> MASTER_LOG_POS=0;
Query OK, 0 rows affected (0.038 sec)

MariaDB [(none)]> START SLAVE;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> SHOW MASTER STATUS;
+-----+-----+-----+-----+
| File          | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000001 | 328 | | |
+-----+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [(none)]> 
```

Il y'a également le fichier de log mysql-bin dans lequel le serveur écrit les modifications sur les bases de données :

```
root@mariadb-2:/var/log/mysql# ls
error.log      error.log.2.gz  error.log.4.gz  error.log.6.gz  mysql-bin.000001
error.log.1.gz error.log.3.gz  error.log.5.gz  error.log.7.gz  mysql-bin.index
root@mariadb-2:/var/log/mysql# 
```

14. Tester la réplication (création de données sur le

maître et vérification sur le slave).

Vérifier que les modifications sur le maître se répliquent sur le slave, il faut aller sur notre serveur maitre maria DB, crée une base de donnée test et une table pour voir si la réplication fonctionne, voir si les donnée et la base de donnée enregistrer vont etre transférer sur mon serveur slave :

```
root@mariadb:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 47
Server version: 10.3.29-MariaDB-0+deb10u1-log Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE test_replication;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> USE test_replication;
Database changed
MariaDB [test_replication]> CREATE TABLE exemple (id INT PRIMARY KEY, nom VARCHAR(50));
Query OK, 0 rows affected (0.041 sec)

MariaDB [test_replication]> INSERT INTO exemple VALUES (1, 'Test');
Query OK, 1 row affected (0.003 sec)

MariaDB [test_replication]> exit
Bye
root@mariadb:~#
```

Nous sommes actuellement sur notre serveur slave et nous pouvons constater que la base de données test a bien été créée et transférée sur notre second serveur et également la table :

```
root@mariadb-2:/var/log/mysql# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 47
Server version: 10.3.29-MariaDB-0+deb10u1-log Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database                |
+-----+
| DOKUWIKI                 |
| glpi                     |
| glpidb                   |
| information_schema       |
| mysql                    |
| performance_schema       |
| test_replication         |
| wordpress_db             |
+-----+
8 rows in set (0.030 sec)

MariaDB [(none)]> 
```

```
MariaDB [(none)]> use test_replication;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [test_replication]> SELECT * FROM exemple;
+-----+-----+
| id | nom |
+-----+-----+
| 1 | Test |
+-----+-----+
1 row in set (0.000 sec)

MariaDB [test_replication]> 
```

Pour un second test nous avons créé une table “testtest” sur une base de donnée test dans notre serveur slave pour voir si la réplication peut fonctionner de SLAVE -> MASTER (ce n’est pas ce qu’on veut)

```

Database changed
MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| test           |
+-----+
1 row in set (0.001 sec)

MariaDB [test]> CREATE TABLE testtest (
  -> id INT AUTO_INCREMENT PRIMARY KEY,
  -> nom VARCHAR(50) NOT NULL,
  -> prenom VARCHAR(50) NOT NULL,
  -> );
ERROR 1064 (42000): You have an error in your SQL syntax; check the
manual that says '...' at line 5
MariaDB [test]> show table;
ERROR 1064 (42000): You have an error in your SQL syntax; check the
manual that says '...' at line 1
MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| test           |
+-----+
1 row in set (0.001 sec)

MariaDB [test]> CREATE TABLE testtest ( id INT AUTO_INCREMENT
ERROR 1064 (42000): You have an error in your SQL syntax; check the
manual that says '...' at line 1
MariaDB [test]> CREATE TABLE testtest ( id INT AUTO_INCREMENT
Query OK, 0 rows affected (0.009 sec)

MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| test           |
| testtest      |
+-----+
2 rows in set (0.001 sec)

MariaDB [test]> _

```

Nous pouvons confirmer que la table “testtest” n’a pas été créée sur notre serveur maître :

```
type help; or \n for help. Type \q to exit
MariaDB [(none)]> use test;
Database changed
MariaDB [test]> CREATE TABLE test (
  -> id INT AUTO_INCREMENT PRIMARY KEY,
  -> nom VARCHAR(50) NOT NULL,
  -> prenom VARCHAR(50) NOT NULL
  -> );
Query OK, 0 rows affected (0.068 sec)

MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| test           |
+-----+
1 row in set (0.001 sec)

MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| test           |
+-----+
1 row in set (0.001 sec)

MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| test           |
+-----+
1 row in set (0.001 sec)

MariaDB [test]> show tables;
+-----+
| Tables_in_test |
+-----+
| test           |
+-----+
1 row in set (0.001 sec)

MariaDB [test]>
```

15. Remonter les logs MySQL/MariaDB sur rsyslog.

Sur le maître et le slave, configurez MySQL pour générer des logs : Ajoutez dans le fichier de configuration MySQL :

general_log_file : spécifie le fichier où les logs seront stockés localement.

general_log : active la génération des logs.

```
[mysqld]
server-id=1 #Identifie de manière unique le m
log_bin=/var/log/mysql/mysql-bin.log # active
general_log_file=/var/log/mysql/general.log
general_log=1
```

Ensuite il faut redémarrer le service mysql :

```

root@mariadb:/etc/mysql# systemctl status mysql
* mariadb.service - MariaDB 10.3.29 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor p
   Active: active (running) since Thu 2024-11-21 10:41:56 CET; 3s ago
     Docs: man:mysql(8)
           https://mariadb.com/kb/en/library/systemd/
   Process: 13373 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d
   Process: 13374 ExecStartPre=/bin/sh -c systemctl unset-environment _WSR
   Process: 13376 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ]
   Process: 13471 ExecStartPost=/bin/sh -c systemctl unset-environment _WSR
   Process: 13473 ExecStartPost=/etc/mysql/debian-start (code=exited, statu
 Main PID: 13439 (mysqld)
    Status: "Taking your SQL requests now..."
     Tasks: 32 (limit: 2322)
    Memory: 106.2M
       CPU: 671ms
    CGroup: /system.slice/mariadb.service
           └─13439 /usr/sbin/mysqld

Nov 21 10:41:56 mariadb systemd[1]: Starting MariaDB 10.3.29 database server
Nov 21 10:41:56 mariadb mysqld[13439]: 2024-11-21 10:41:56 0 [Note] /usr/sb
Nov 21 10:41:56 mariadb systemd[1]: Started MariaDB 10.3.29 database server
Nov 21 10:41:56 mariadb /etc/mysql/debian-start[13478]: /usr/bin/mysql_upgra
Nov 21 10:41:56 mariadb /etc/mysql/debian-start[13478]: Looking for 'mysql'
Nov 21 10:41:56 mariadb /etc/mysql/debian-start[13478]: Looking for 'mysqld
Nov 21 10:41:56 mariadb /etc/mysql/debian-start[13478]: This installation of
Nov 21 10:41:56 mariadb /etc/mysql/debian-start[13486]: Checking for insecure
Nov 21 10:41:56 mariadb debian-start[13492]: WARNING: tempfile is deprecated
lines 1-27/27 (END)

```

Il faut ensuite toujours sur notre machine mariadb aller sur le fichier de configuration rsyslog.conf qui se trouve dans **/etc/rsyslog.conf** et vérifier les packet UDP et TCP sont bien décocher et dans le bon port :

```

# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html

#####
#### MODULES ####
#####

module(load="imuxsock") # provides support for local system logging
module(load="imklog") # provides kernel logging support
#module(load="immark") # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="imtcp")
input(type="imtcp" port="514")

```

Ensuite il faut rajouter cette ligne pour envoyer les logs au serveur Rsyslog centralisé :

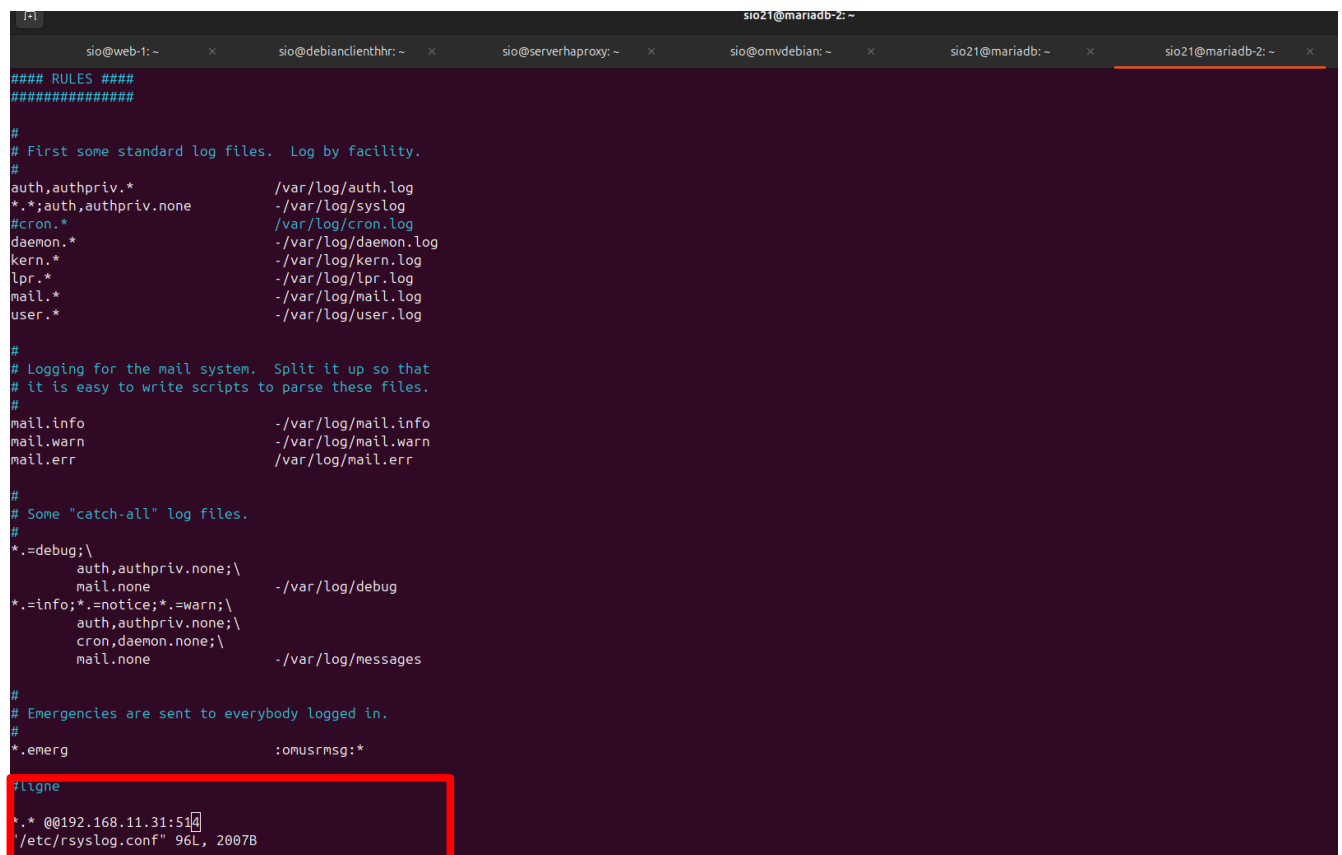
. : indique que tous les types de logs doivent être envoyés.

@ : désigne un envoi via UDP.

192.168.11.31:514 : IP et port du serveur Rsyslog qui recevra les logs.

```
*.* @@192.168.11.31:514
"rsyslog.conf" 96L, 2007B
```

Il faut également faire pareil pour mariadb-2 : le serveur slave :



```
##### RULES #####
#####

#
# First some standard log files.  Log by facility.
#
auth,authpriv.*          /var/log/auth.log
*.*;auth,authpriv.none  -/var/log/syslog
#cron.*                  /var/log/cron.log
daemon.*                 /var/log/daemon.log
kern.*                   /var/log/kern.log
lpr.*                    /var/log/lpr.log
mail.*                   /var/log/mail.log
user.*                   /var/log/user.log

#
# Logging for the mail system.  Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info                /var/log/mail.info
mail.warn                /var/log/mail.warn
mail.err                 /var/log/mail.err

#
# Some "catch-all" log files.
#
*.*=debug;\
    auth,authpriv.none;\
    mail.none            /var/log/debug
*.*=info;*.=notice;*.=warn;\
    auth,authpriv.none;\
    cron,daemon.none;\
    mail.none            /var/log/messages

#
# Emergencies are sent to everybody logged in.
#
*.emerg                  :omusrmsg:*

#Tlgne
*.* @@192.168.11.31:514
"/etc/rsyslog.conf" 96L, 2007B
```

Nous pouvons enfin confirmer la réception de nos log depuis le serveur rsyslog et dans le fichier de conf :

/var/log/syslog.log

```
2024-11-21T10:52:10+01:00 mariadb systemd[1]: Started MariaDB 10.3.29 database server.
2024-11-21T10:52:10+01:00 mariadb /etc/mysql/debian-start[13685]: Upgrading MySQL tables if necessary.
2024-11-21T10:52:10+01:00 mariadb /etc/mysql/debian-start[13688]: /usr/bin/mysql_upgrade: the '--basedir' option is always ignored
2024-11-21T10:52:10+01:00 mariadb /etc/mysql/debian-start[13688]: Looking for 'mysql' as: /usr/bin/mysql
2024-11-21T10:52:10+01:00 mariadb /etc/mysql/debian-start[13688]: Looking for 'mysqlcheck' as: /usr/bin/mysqlcheck
2024-11-21T10:52:10+01:00 mariadb /etc/mysql/debian-start[13688]: This installation of MySQL is already upgraded to 10.3.29-MariaDB, use --force if you still need to run mysql_upgrade
2024-11-21T10:52:10+01:00 mariadb /etc/mysql/debian-start[13696]: Checking for insecure root accounts.
2024-11-21T10:52:10+01:00 mariadb /etc/mysql/debian-start[13700]: Triggering myisam-recover for all MyISAM tables and aria-recover for all Aria tables
2024-11-21T10:52:10+01:00 mariadb debian-start[13702]: WARNING: tempfile is deprecated; consider using mktemp instead.
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Stopping System Logging Service...
2024-11-21T10:52:17+01:00 mariadb rsyslogd: [origin software="rsyslogd" swVersion="8.2102.0" x-pid="7820" x-info="https://www.rsyslog.com"] exiting on signal 15.
2024-11-21T10:52:17+01:00 mariadb systemd[1]: rsyslog.service: Succeeded.
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Stopped System Logging Service.
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Starting System Logging Service...
2024-11-21T10:52:17+01:00 mariadb rsyslogd: imuxsock: Acquired UNIX socket '/run/systemd/journal/syslog' (fd 3) from systemd. [v8.2102.0]
2024-11-21T10:52:17+01:00 mariadb rsyslogd: [origin software="rsyslogd" swVersion="8.2102.0" x-pid="13771" x-info="https://www.rsyslog.com"] start
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Started System Logging Service.
2024-11-21T10:52:22+01:00 mariadb qemu-ga: info: guest-ping called
2024-11-21T10:54:03+01:00 192.168.13.20 mgr: SME TELNET from 192.168.13.7 - MANAGER Mode
2024-11-21T10:54:03+01:00 mariadb-2 systemd[1]: Stopping MariaDB 10.3.29 database server...
2024-11-21T10:54:04+01:00 mariadb-2 systemd[1]: mariadb.service: Succeeded.
2024-11-21T10:54:04+01:00 mariadb-2 systemd[1]: Stopped MariaDB 10.3.29 database server.
2024-11-21T10:54:04+01:00 mariadb-2 systemd[1]: mariadb.service: Consumed 2.814s CPU time.
2024-11-21T10:54:04+01:00 mariadb-2 systemd[1]: Starting MariaDB 10.3.29 database server...
2024-11-21T10:54:04+01:00 mariadb-2 mysqld[1003]: 2024-11-21 10:54:04 0 [Note] /usr/sbin/mysqld (mysqld 10.3.29-MariaDB-0+deb10u1-log) starting as process 1003 ...
2024-11-21T10:54:05+01:00 mariadb-2 systemd[1]: Started MariaDB 10.3.29 database server.
2024-11-21T10:54:05+01:00 mariadb-2 /etc/mysql/debian-start[1041]: Upgrading MySQL tables if necessary.
2024-11-21T10:54:05+01:00 mariadb-2 /etc/mysql/debian-start[1044]: /usr/bin/mysql_upgrade: the '--basedir' option is always ignored
2024-11-21T10:54:05+01:00 mariadb-2 /etc/mysql/debian-start[1044]: Looking for 'mysql' as: /usr/bin/mysql
2024-11-21T10:54:05+01:00 mariadb-2 /etc/mysql/debian-start[1044]: Looking for 'mysqlcheck' as: /usr/bin/mysqlcheck
2024-11-21T10:54:05+01:00 mariadb-2 /etc/mysql/debian-start[1044]: This installation of MySQL is already upgraded to 10.3.29-MariaDB, use --force if you still need to run mysql_upgrade
2024-11-21T10:54:05+01:00 mariadb-2 /etc/mysql/debian-start[1053]: Checking for insecure root accounts.
2024-11-21T10:54:05+01:00 mariadb-2 /etc/mysql/debian-start[1057]: Triggering myisam-recover for all MyISAM tables and aria-recover for all Aria tables
2024-11-21T10:54:05+01:00 mariadb-2 debian-start[1059]: WARNING: tempfile is deprecated; consider using mktemp instead.
2024-11-21T10:54:09+01:00 mariadb-2 systemd[1]: Stopping System Logging Service...
2024-11-21T10:54:09+01:00 mariadb-2 rsyslogd: [origin software="rsyslogd" swVersion="8.2102.0" x-pid="409" x-info="https://www.rsyslog.com"] exiting on signal 15.
2024-11-21T10:54:09+01:00 mariadb-2 systemd[1]: rsyslog.service: Succeeded.
2024-11-21T10:54:09+01:00 mariadb-2 systemd[1]: Stopped System Logging Service.
2024-11-21T10:54:09+01:00 mariadb-2 systemd[1]: Starting System Logging Service...
2024-11-21T10:54:09+01:00 mariadb-2 rsyslogd: imuxsock: Acquired UNIX socket '/run/systemd/journal/syslog' (fd 3) from systemd. [v8.2102.0]
2024-11-21T10:54:09+01:00 mariadb-2 rsyslogd: [origin software="rsyslogd" swVersion="8.2102.0" x-pid="1128" x-info="https://www.rsyslog.com"] start
2024-11-21T10:54:09+01:00 mariadb-2 systemd[1]: Started System Logging Service.
root@serv-syslog:~#
```

```
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Stopping System Logging Service...
2024-11-21T10:52:17+01:00 mariadb rsyslogd: [origin software="rsyslogd" swVersion="8.2102.0" x-pid="7820" x-info="https://www.rsyslog.com"] exiting on signal 15.
2024-11-21T10:52:17+01:00 mariadb systemd[1]: rsyslog.service: Succeeded.
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Stopped System Logging Service.
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Starting System Logging Service...
2024-11-21T10:52:17+01:00 mariadb rsyslogd: imuxsock: Acquired UNIX socket '/run/systemd/journal/syslog' (fd 3) from systemd. [v8.2102.0]
2024-11-21T10:52:17+01:00 mariadb rsyslogd: [origin software="rsyslogd" swVersion="8.2102.0" x-pid="13771" x-info="https://www.rsyslog.com"] start
2024-11-21T10:52:17+01:00 mariadb systemd[1]: Started System Logging Service.
2024-11-21T10:52:22+01:00 mariadb qemu-ga: info: guest-ping called
2024-11-21T10:54:03+01:00 192.168.13.20 mgr: SME TELNET from 192.168.13.7 - MANAGER Mode
2024-11-21T10:54:03+01:00 mariadb-2 systemd[1]: Stopping MariaDB 10.3.29 database server...
2024-11-21T10:54:04+01:00 mariadb-2 systemd[1]: mariadb.service: Succeeded.
2024-11-21T10:54:04+01:00 mariadb-2 systemd[1]: Stopped MariaDB 10.3.29 database server.
```

16. Configurer le fichier haproxy.cfg (frontend, backend, etc.).

Une fois que c'est fait, installez HAProxy en utilisant la commande suivante :

```
sudo apt install haproxy -y
```

et vérifier que haproxy est installé avec la commande :
`dpkg -l | grep haproxy` ou `haproxy -v`

```
Installation des actions différées (« triggers ») pour man-db (2.11.2-2) ...
root@serverhaproxy:/etc# dpkg -l | grep haproxy/
root@serverhaproxy:/etc# dpkg -l | grep haproxy
ii haproxy                2.6.12-1+deb12u1          amd64          fast and reliable load balancing
reverse proxy
root@serverhaproxy:/etc#
```

```
root@serverhaproxy:/etc# haproxy -v
HAProxy version 2.6.12-1+deb12u1 2023/12/16 - https://haproxy.org/
Status: long-term supported branch - will stop receiving fixes around Q2 2027.
Known bugs: http://www.haproxy.org/bugs/bugs-2.6.12.html
Running on: Linux 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64
root@serverhaproxy:/etc#
```

Voici le contenu du fichier de configuration de haproxy qui se trouve dans : `/etc/haproxy/haproxy.cfg` il faudra

ensuite déclarer le back end et frond end :

```
global
log /dev/log      local0
log /dev/log      local1 notice
chroot /var/lib/haproxy
stats socket /run/haproxy/admin.sock mode 660 level admin
stats timeout 30s
user haproxy
group haproxy
daemon

# Default SSL material locations
ca-base /etc/ssl/certs
crt-base /etc/ssl/private

# See: https://ssl-config.mozilla.org/#server=haproxy&server-version=2.0
ssl-default-bind-ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-
HACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384
ssl-default-bind-ciphersuites TLS_AES_128_GCM_SHA256:TLS_AES_256_GCM_SHA
ssl-default-bind-options ssl-min-ver TLSv1.2 no-tls-tickets

defaults
log      global
mode    http
option  httplog
option  dontlognull
timeout connect 5000
timeout client 50000
timeout server 50000
errorfile 400 /etc/haproxy/errors/400.http
errorfile 403 /etc/haproxy/errors/403.http
errorfile 408 /etc/haproxy/errors/408.http
errorfile 500 /etc/haproxy/errors/500.http
errorfile 502 /etc/haproxy/errors/502.http
errorfile 503 /etc/haproxy/errors/503.http
errorfile 504 /etc/haproxy/errors/504.http
```

Voici le status du service haproxy :

```

root@serverhaproxy:/etc# systemctl status haproxy
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-11-21 11:03:05 CET; 2min 43s ago
     Docs: man:haproxy(1)
           file:/usr/share/doc/haproxy/configuration.txt.gz
   Main PID: 4542 (haproxy)
      Tasks: 2 (limit: 2306)
     Memory: 45.7M
        CPU: 138ms
    CGroup: /system.slice/haproxy.service
            └─4542 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
              └─4544 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

nov. 21 11:03:05 serverhaproxy systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
nov. 21 11:03:05 serverhaproxy haproxy[4542]: [NOTICE] (4542) : New worker (4544) forked
nov. 21 11:03:05 serverhaproxy haproxy[4542]: [NOTICE] (4542) : Loading success.
nov. 21 11:03:05 serverhaproxy systemd[1]: Started haproxy.service - HAProxy Load Balancer.
root@serverhaproxy:/etc#

```

FRONTEND : Ici, nous allons déclarer un frontend nommé "front_webservers" accessible sur le port 80. Il sera associé au backend nommé "backend_webservers". L'option "forwardfor" sera déclarée pour que l'en-tête "X-Forwarded-For" soit ajoutée aux requêtes HTTP afin que le serveur web ait connaissance de l'adresse IP d'origine du client (sinon, il verra toujours l'adresse IP du HAProxy).

BACKEND : Cette section définit les serveurs vers lesquels HAProxy redirigera le trafic correspondant aux flux à destination du frontend "front_webservers", puisqu'il est associé au backend que nous configurer.

HAProxy va distribuer le trafic entre les deux serveurs

web nommés "web1" et "web2" et "web3". La méthode d'équilibrage de charge "roundrobin" sera utilisée, ce qui signifie que HAProxy alternera entre chaque serveur web. Dans l'exemple ci-dessous, vous devez penser à adapter les adresses IP des serveurs backends, et éventuellement, les noms.

```
frontend front_web_servers
    bind *:80
    default_backend backend_webservers
    option forwardfor

backend backend_webservers
    balance roundrobin
    server web1 192.168.11.32:80 check
    server web2 192.168.11.33:80 check
    server web3 192.168.11.34:80 check
```

J'avais eu une erreur et le statut du service me la fais savoir (j'avais mis le mauvais port") mais dans le screen au dessus c'est le bon j'avais modifié :

```
L'unité (unit) haproxy.service a commencé à démarrer.
nov. 21 11:10:19 serverhaproxy haproxy[4685]: [NOTICE] (4685) : haproxy version is 2.6.12-1+deb12u1
nov. 21 11:10:19 serverhaproxy haproxy[4685]: [NOTICE] (4685) : path to executable is /usr/sbin/haproxy
nov. 21 11:10:19 serverhaproxy haproxy[4685]: [ALERT] (4685) : config : parsing [/etc/haproxy/haproxy.cfg:38] : unknown keyword 'default_backend' in 'frontend' section;
nov. 21 11:10:19 serverhaproxy haproxy[4685]: [ALERT] (4685) : config : Error(s) found in configuration file : /etc/haproxy/haproxy.cfg
nov. 21 11:10:19 serverhaproxy haproxy[4685]: [ALERT] (4685) : config : Fatal errors found in configuration.
nov. 21 11:10:19 serverhaproxy systemd[1]: haproxy.service: Main process exited, code=exited, status=1/FAILURE
Subject: Unit process exited
```

Maintenant le service est OK :

```

root@serverhaproxy:/etc# systemctl status haproxy
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-11-21 11:19:43 CET; 2s ago
     Docs: man:haproxy(1)
           file:/usr/share/doc/haproxy/configuration.txt.gz
   Main PID: 4711 (haproxy)
      Tasks: 2 (limit: 2306)
     Memory: 37.7M
        CPU: 105ms
    CGroup: /system.slice/haproxy.service
            └─4711 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
              └─4714 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

nov. 21 11:19:43 serverhaproxy systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
nov. 21 11:19:43 serverhaproxy haproxy[4711]: [NOTICE] (4711) : New worker (4714) forked
nov. 21 11:19:43 serverhaproxy haproxy[4711]: [NOTICE] (4711) : Loading success.
nov. 21 11:19:43 serverhaproxy systemd[1]: Started haproxy.service - HAProxy Load Balancer.
root@serverhaproxy:/etc#

```

17. Tester la répartition de charge (arrêter un serveur web et voir si l'autre prend le relais).

Pour ce faire il faut aller sur une machine serveur web 1 des 3 et voir si apache fonctionne dans ce cas il fonctionne correctement, il faudra le stopper pour rendre le serveur inactif :

```
root@web-1:/usr/lib/naaios/plugins# systemctl status apache2
○ apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
  Active: inactive (dead) since Thu 2024-11-21 11:21:51 CET; 3s ago
  Duration: 1d 10h 10min 11.123s
  Docs: https://httpd.apache.org/docs/2.4/
  Process: 9351 ExecStop=/usr/sbin/apachectl graceful-stop (code=exited, status=0/SUCCESS)
  Main PID: 805 (code=exited, status=0/SUCCESS)
  CPU: 9.004s

nov. 21 00:00:00 web-1 systemd[1]: Reloading apache2.service - The Apache HTTP Server...
nov. 21 00:00:00 web-1 apachectl[4614]: AH00557: apache2: apr_sockaddr_info_get() failed for web-1
nov. 21 00:00:00 web-1 apachectl[4614]: AH00558: apache2: Could not reliably determine the server
nov. 21 00:00:00 web-1 systemd[1]: Reloaded apache2.service - The Apache HTTP Server.
nov. 21 11:21:51 web-1 systemd[1]: Stopping apache2.service - The Apache HTTP Server...
nov. 21 11:21:51 web-1 apachectl[9353]: AH00557: apache2: apr_sockaddr_info_get() failed for web-1
nov. 21 11:21:51 web-1 apachectl[9353]: AH00558: apache2: Could not reliably determine the server
nov. 21 11:21:51 web-1 systemd[1]: apache2.service: Deactivated successfully.
nov. 21 11:21:51 web-1 systemd[1]: Stopped apache2.service - The Apache HTTP Server.
nov. 21 11:21:51 web-1 systemd[1]: apache2.service: Consumed 9.004s CPU time.
lines 1-19/19 (END)
```

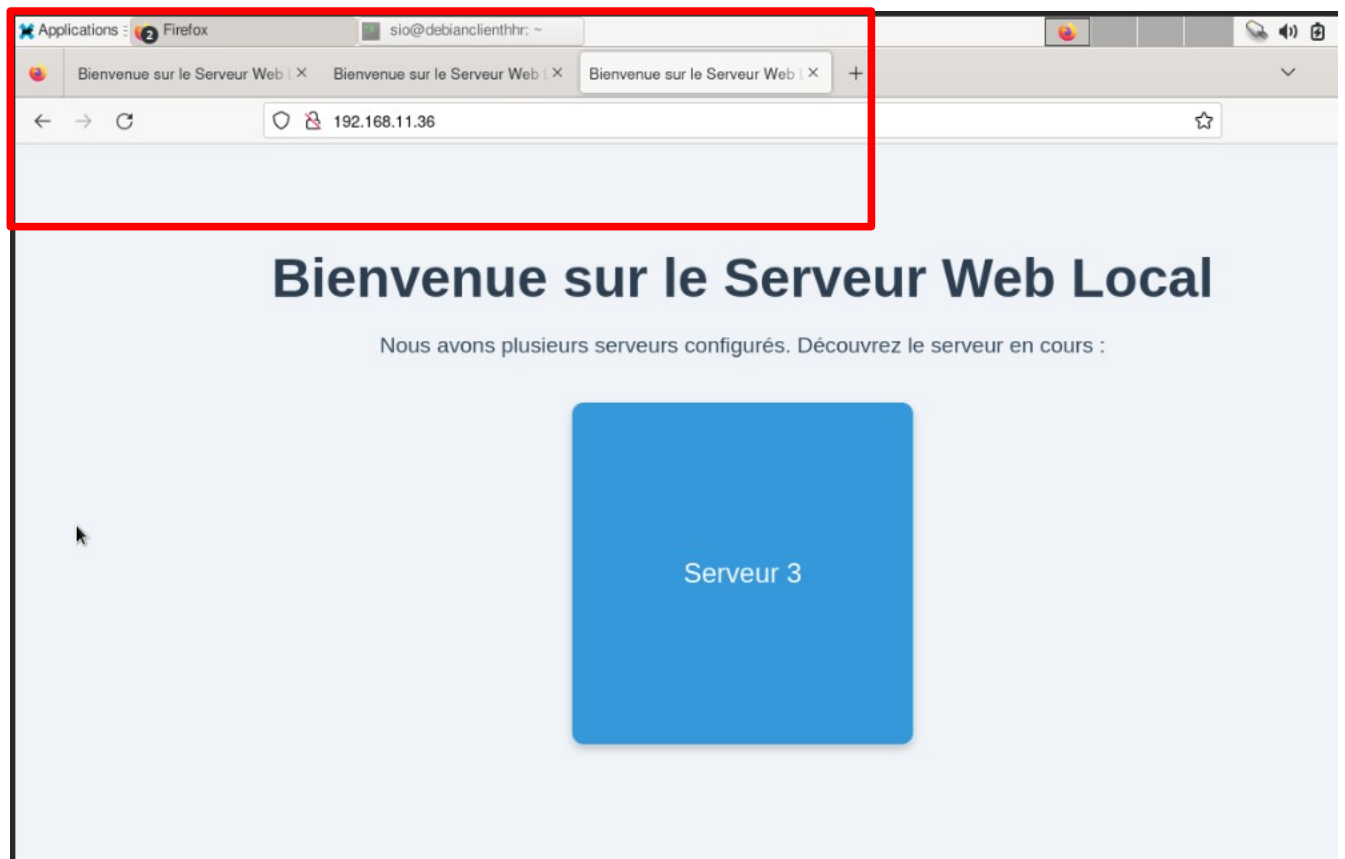
Dans le serveur haproxy le statut du service nous montre que le serveur 1 est down, c'est une bonne nouvelle :

```
sio@web-1:~ x sio@debianclienthr:~ x sio@serverhaproxy:~ x sio@omidebian:~ x sio21@mariadb:~ x sio21@mariadb-2:~ x sio@serv-syslo
haproxy.service - HAProxy Load Balancer
  Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
  Active: active (running) since Thu 2024-11-21 11:19:43 CET; 8min ago
  Docs: man:haproxy(1)
  file:/usr/share/doc/haproxy/configuration.txt.gz
  Main PID: 4711 (haproxy)
  Tasks: 2 (limit: 2306)
  Memory: 41.7M
  CPU: 336ms
  CGroup: /system.slice/haproxy.service
  └─4711 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
  └─4714 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

nov. 21 11:21:57 serverhaproxy haproxy[4714]: Server backend_webservers/web1 is DOWN, reason: Layer4 connection problem, info: "Connection refused", check duration: 0ms. 2 active
nov. 21 11:24:57 serverhaproxy haproxy[4714]: 192.168.11.251:53578 [21/Nov/2024:11:24:57.754] front_web_servers backend_webservers/web2 0/0/0/2/2 200 1105 - - - - 1/1/0/0/0 0/0
nov. 21 11:24:57 serverhaproxy haproxy[4714]: 192.168.11.251:53578 [21/Nov/2024:11:24:57.892] front_web_servers backend_webservers/web3 0/0/1/1/2 404 435 - - - - 1/1/0/0/0 0/0
nov. 21 11:25:22 serverhaproxy haproxy[4714]: [WARNING] (4714) : Server backend_webservers/web1 is UP, reason: Layer4 check passed, check duration: 0ms. 3 active and 0 backup servers online. 0 sess
nov. 21 11:25:22 serverhaproxy haproxy[4714]: Server backend_webservers/web1 is UP, reason: Layer4 check passed, check duration: 0ms. 3 active and 0 backup servers online. 0 sess
nov. 21 11:25:28 serverhaproxy haproxy[4714]: 192.168.11.251:53578 [21/Nov/2024:11:25:28.223] front_web_servers backend_webservers/web2 0/0/0/2/2 200 1105 - - - - 1/1/0/0/0 0/0
nov. 21 11:25:29 serverhaproxy haproxy[4714]: 192.168.11.251:53578 [21/Nov/2024:11:25:29.384] front_web_servers backend_webservers/web3 0/0/1/1/2 200 1106 - - - - 1/1/0/0/0 0/0
nov. 21 11:25:31 serverhaproxy haproxy[4714]: 192.168.11.251:53578 [21/Nov/2024:11:25:31.511] front_web_servers backend_webservers/web1 0/0/1/1/2 200 1106 - - - - 1/1/0/0/0 0/0
nov. 21 11:26:53 serverhaproxy haproxy[4714]: 192.168.11.251:47456 [21/Nov/2024:11:26:53.202] front_web_servers backend_webservers/web2 0/0/0/1/1 200 1105 - - - - 1/1/0/0/0 0/0
```

il faut ensuite depuis un navigateur tester avec l'adresse ip de haproxy en l'occurrence c'est l'ip **192.168.11.36**, nous pouvons constater que sa n'affiche pas le serveur 1 mais bien le serveur 2 et 3 :





18. Configurer les statistiques HaProxy.

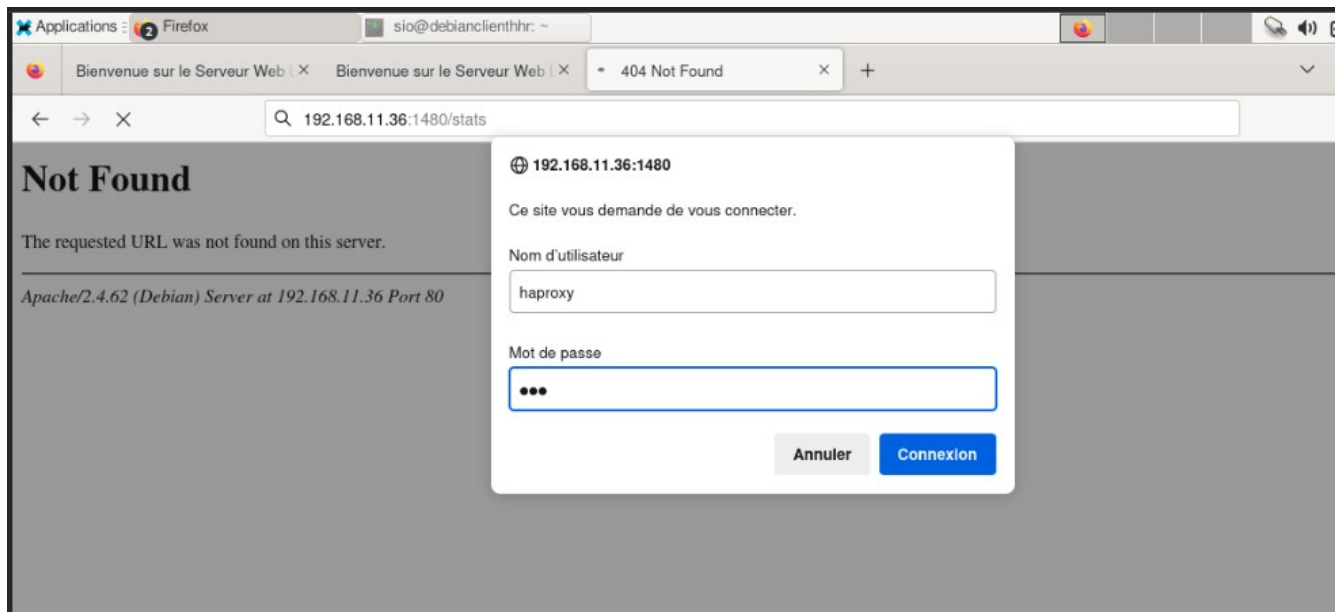
HAProxy embarque une interface prête à l'emploi permettant d'avoir une vue d'ensemble sur les frontends et les serveurs de backends, avec un ensemble de statistiques. De plus, cette page indique l'état de chaque serveur de backend.

Pour activer et configurer cette page de statistiques, vous devez éditer le fichier de configuration "haproxy.cfg". Puis, ajoutez ceci à la fin du fichier, à la suite de la configuration précédente, pour déclarer le frontend "front_stats" :

```
frontend front_stats
    bind *:1480
    stats enable
    stats uri /stats
    stats hide-version
    stats refresh 10s
    stats realm Haproxy\ Statistics
    stats auth haproxy:sio
```

L'interface sera accessible sur le port "1480" (bind) en indiquant l'adresse IP du serveur HAProxy suivi de "/stats" (stats uri). Les statistiques de cette page seront actualisées toutes les 10 secondes (stats refresh) et elle sera protégée par un identifiant et un mot de passe (stats auth). Dans l'exemple ci-dessus, l'utilisateur s'appelle "haproxy" et le mot de passe est "sio".

Enregistrez la configuration, redémarrez HAProxy et tentez d'accéder à la page suivante : "http://192.168.11.36/stats". Après avoir saisi les identifiants, vous devriez arriver sur cette page :



HAProxy

Statistics Report for pid 4782

> General process information

pid = 4782 (process #1, nproc = 1, nbthread = 1)
 uptime = 0d 0h02m41s
 system limits: memmax = unlimited; ulimit-n = 524287
 maxsock = 524287; maxconn = 262123; maxpipes = 0
 current conns = 1; current pipes = 0/0; conn rate = 0/sec; bit rate = 0.000 kbps
 Running tasks: 0/17; idle = 100 %

active UP
 active UP, going down
 active DOWN, going up
 active or backup DOWN
 active or backup DOWN for maintenance (MAINT)
 active or backup SOFT STOPPED for maintenance

backup UP
 backup UP, going down
 backup DOWN, going up
 not checked

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

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

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

front_web_servers																																
	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server											
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle		
Frontend				0	1	-	0	2		262	123	2	1	793	2	175	0	0	0	0	0	OPEN										
backend_webservers																																
	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server											
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle		
web1	0	0	-	0	1	0	0	1	-	2	2	1m41s	706	870	0	0	0	0	0	0	0	2m41s UP	L4OK in 0ms	1/1	Y	-	0	0	0s	-		
web2	0	0	-	0	1	0	0	1	-	2	2	1m27s	711	870	0	0	0	0	0	0	0	2m41s UP	L4OK in 0ms	1/1	Y	-	0	0	0s	-		
web3	0	0	-	0	1	0	0	1	-	1	1	1m46s	376	435	0	0	0	0	0	0	0	2m41s UP	L4OK in 0ms	1/1	Y	-	0	0	0s	-		
Backend	0	0	-	0	1	0	0	1	26	213	5	5	1m27s	1	793	2	175	0	0	0	0	0	2m41s UP		3/3	3	0	0	0	0s	-	
front_stats																																
	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings		Server											
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle		
Frontend				0	1	-	1	1		262	123	4	1	834	49	246	0	0	0	0	0	OPEN										

Je vais vous détailler quelques résultats obtenus : La section Taux de session, dans le contexte d'une interface, décrit la vitesse à laquelle les clients se connectent à HAProxy.

Session rate		
Cur	Max	Limit
0	0	100

La section Sessions compte le nombre de sessions, ou de connexions client-serveur complètes, utilisées sur l'équilibreur de charge.

Sessions					
Cur	Max	Limit	Total	LbTot	Last
0	0	20 000	0		

La section Bytes affiche la quantité cumulée de données envoyées et reçues entre HAProxy et les clients en aval :

Bytes	
In	Out
15 305	30 514

- Ajouter HaProxy à la supervision Nagios (état, charge, interfaces, etc.).**

Installation de l'NRPE et de nagios plugins sur ma machine haproxy, il faut ensuite modifier le fichier de conf de nagios `/etc/nagios/nrpe.cfg` et mettre une ALLOWING ip dont celle de nagios :

```
root@serverhaproxy:/etc# systemctl status nagios-nrpe-server.service
● nagios-nrpe-server.service - Nagios Remote Plugin Executor
   Loaded: loaded (/lib/systemd/system/nagios-nrpe-server.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-11-21 11:58:56 CET; 16s ago
     Docs: http://www.nagios.org/documentation
  Main PID: 9003 (nrpe)
    Tasks: 1 (limit: 2306)
   Memory: 1.2M
      CPU: 13ms
   CGroup: /system.slice/nagios-nrpe-server.service
           └─9003 /usr/sbin/nrpe -c /etc/nagios/nrpe.cfg -f

nov. 21 11:58:56 serverhaproxy systemd[1]: Started nagios-nrpe-server.service - Nagios Remote Plugin Executor.
nov. 21 11:58:56 serverhaproxy nrpe[9003]: Starting up daemon
nov. 21 11:58:56 serverhaproxy nrpe[9003]: Server listening on 0.0.0.0 port 5666.
nov. 21 11:58:56 serverhaproxy nrpe[9003]: Server listening on :: port 5666.
nov. 21 11:58:56 serverhaproxy nrpe[9003]: Listening for connections on port 5666
nov. 21 11:58:56 serverhaproxy nrpe[9003]: Allowing connections from: 127.0.0.1,192.168.13.8
root@serverhaproxy:/etc#
```

```
nov. 21 05:07:00 srv-nagios nagios[300300]: successfully launched command file worker with pid 303313
root@srv-nagios:/usr/local/nagios/etc/servers# /usr/local/nagios/libexec/check_nrpe -H 192.168.11.36
NRPE v4.1.0
root@srv-nagios:/usr/local/nagios/etc/servers#
```

Ensuite il faut définir les services comme l'état du system, la charge, le disque local, le taux de remplissage du disque dur, et le ping status :

```

# Nagios Host configuration file template
define host {
    use                linux-server
    host_name          serverhaproxy
    alias              server ha proxy
    address            192.168.11.36
    register           1
}

define service {
    use                local-service
    host_name          serverhaproxy
    service_description PING STATUS
    check_command      check_ping!100.0,20%!500.0,60%
}

define service {
    use                local-service
    host_name          serverhaproxy
    service_description Load Average
    check_command      check_local_load!15,10,5!20,15,10
}

define service {
    use                local-service
    host_name          serverhaproxy
    service_description Local Disk
    check_command      check_local_disk!20%!10%!/
}

define service {
    use                local-service
    host_name          serverhaproxy
    service_description Etat du systeme
    check_command      check-host-alive
}

define service {
    use                local-service
    host_name          serverhaproxy

```

```

define service {
    use                local-service
    host_name          serverhaproxy
    service_description Etat du systeme
    check_command       check-host-alive
}

define service {
    use                local-service
    host_name          serverhaproxy
    service_description Charge systeme
    check_command       check_local_load
}

define service {
    use                local-service
    host_name          serverhaproxy
    service_description Taux de remplissage du disque dur
    check_command       check_local_disk!20%!10%!/
}

```

serverhaproxy	Charge systeme	UNKNOWN	11-21-2024 16:36:16	0d 1h 33m 56s	4/4	Le seul d'alerte doit être un nombre à virgule flottante!
	Etat du systeme	OK	11-21-2024 16:38:12	0d 1h 45m 55s	1/4	PING OK - Paquets perdus = 0%, RTA = 1.01 ms
	Load Average	OK	11-21-2024 16:39:32	0d 2h 9m 40s	1/4	OK - Charge moyenne: 0.03, 0.02, 0.00
	Local Disk	OK	11-21-2024 16:35:52	0d 2h 7m 20s	1/4	DISK OK - free space: / 26779 MB (90,74% inode=97%):
	PING STATUS	OK	11-21-2024 16:35:30	0d 0h 15m 46s	1/4	PING OK - Paquets perdus = 0%, RTA = 1.25 ms
	Taux de remplissage du disque dur	OK	11-21-2024 16:38:22	0d 1h 31m 50s	1/4	DISK OK - free space: / 26779 MB (90,74% inode=97%):
serverhaproxy2	Charge systeme	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:41:21 CET 2024
	Etat du systeme	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:42:38 CET 2024
	Load Average	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:43:55 CET 2024
	Local Disk	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:44:04 CET 2024
	PING STATUS	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:41:29 CET 2024
	Taux de remplissage du disque dur	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:42:46 CET 2024

20. Remonter les logs HaProxy sur rsyslog.

Pour la première étape il faut vérifier que rsyslog est bien installé sur ma machine qui doit envoyer les logs donc il faut regarder le statut du service rsyslog :

```
root@serverhaproxy:/etc# systemctl status rsyslog.service
● rsyslog.service - System Logging Service
   Loaded: loaded (/lib/systemd/system/rsyslog.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-11-21 15:56:31 CET; 33s ago
 TriggeredBy: ● syslog.socket
   Docs: man:rsyslogd(8)
         man:rsyslog.conf(5)
         https://www.rsyslog.com/doc/
   Main PID: 9253 (rsyslogd)
     Tasks: 4 (limit: 2306)
    Memory: 1.4M
       CPU: 10ms
    CGroup: /system.slice/rsyslog.service
           └─9253 /usr/sbin/rsyslogd -n -iNONE

nov. 21 15:56:31 serverhaproxy systemd[1]: Starting rsyslog.service - System Logging Service...
nov. 21 15:56:31 serverhaproxy systemd[1]: Started rsyslog.service - System Logging Service.
nov. 21 15:56:31 serverhaproxy rsyslogd[9253]: imuxsock: Acquired UNIX socket '/run/systemd/journal/syslog' (fd 3) from systemd. [v8.2302.0]
nov. 21 15:56:31 serverhaproxy rsyslogd[9253]: [origin software="rsyslogd" swVersion="8.2302.0" x-pid="9253" x-info="https://www.rsyslog.com"] start
root@serverhaproxy:/etc#
```

Il faut ensuite configurer le fichier `/etc/rsyslog.conf` de ma machine haproxy en ajoutant la ligne `*.*@@192.168.11.31:514` qui va permettre de envoyer les logs vers le serveur central (serversyslog)

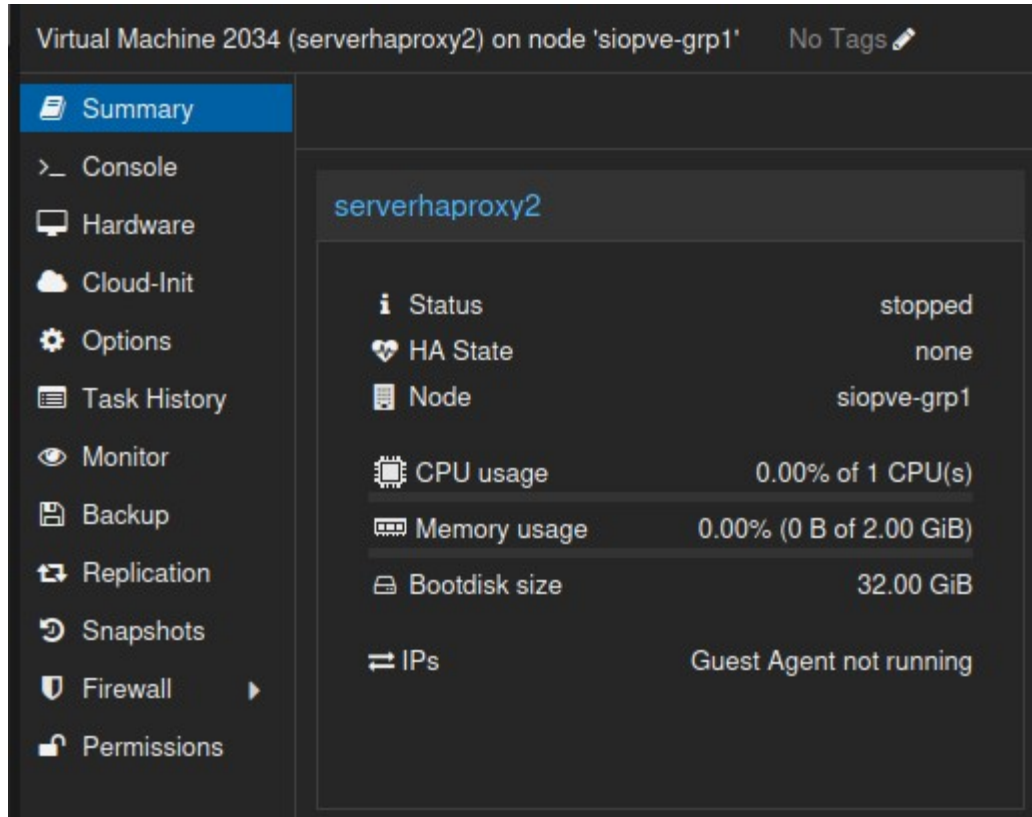
Depuis notre serveur rsyslog, nous pouvons voir les logs

de haproxy, par exemple quand j'ai activé ou stoppé le statut de rsyslog ou haproxy :

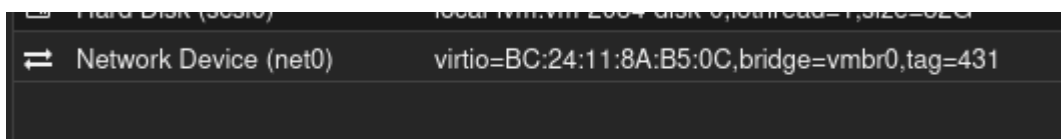
```
2024-11-21T15:58:50+01:00 serverhaproxy rsyslogd: [origin software="rsyslogd" swVersion="8.2302.0" x-pid="9253" x-info="https://www.rsyslog.com"] exit
2024-11-21T15:58:50+01:00 serverhaproxy systemd[1]: Stopping rsyslog.service - System Logging Service...
2024-11-21T15:58:50+01:00 serverhaproxy systemd[1]: rsyslog.service: Deactivated successfully.
2024-11-21T15:58:50+01:00 serverhaproxy systemd[1]: Stopped rsyslog.service - System Logging Service.
2024-11-21T15:58:50+01:00 serverhaproxy systemd[1]: Starting rsyslog.service - System Logging Service...
2024-11-21T15:58:50+01:00 serverhaproxy systemd[1]: Started rsyslog.service - System Logging Service.
2024-11-21T15:58:50+01:00 serverhaproxy rsyslogd: unixsock: Acquired UNIX socket '/run/systemd/journal/syslog' (fd 3) from systemd. [v8.2302.0]
2024-11-21T15:58:50+01:00 serverhaproxy rsyslogd: [origin software="rsyslogd" swVersion="8.2302.0" x-pid="9308" x-info="https://www.rsyslog.com"] started
2024-11-21T16:00:01+01:00 srvwebwordpress CRON[102059]: (www-data) CMD ([ -x /usr/share/awstats/tools/update.sh ] && /usr/share/awstats/tools/update.sh)
2024-11-21T16:00:02+01:00 192.168.13.20 mgr: SME TELNET from 192.168.13.7 - MANAGER Mode
2024-11-21T16:00:01+01:00 srvwebwordpress CRON[102058]: (CRON) info (No MTA installed, discarding output)
2024-11-21T16:00:32+01:00 serverhaproxy systemd[1]: Stopping haproxy.service - HAProxy Load Balancer...
2024-11-21T16:00:32+01:00 serverhaproxy systemd[1]: haproxy.service: Deactivated successfully.
2024-11-21T16:00:32+01:00 serverhaproxy systemd[1]: Stopped haproxy.service - HAProxy Load Balancer.
2024-11-21T16:00:32+01:00 serverhaproxy systemd[1]: haproxy.service: Consumed 6.076s CPU time.
2024-11-21T16:00:32+01:00 serverhaproxy systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
2024-11-21T16:00:32+01:00 serverhaproxy systemd[1]: Started haproxy.service - HAProxy Load Balancer.
root@serverhaproxy:~#
```

21. Cloner le serveur HaProxy, modifier son hostname et IP.

Nous avons cloné une VM linux serveur avec un ID 2034 et un nom spécifique :



Ne pas oublier de mettre l'id du VLAN 431 (LAN)



Le hostname à été modifié mais également l'interface réseau qui est en LAN :

```
root@serverhaproxy2:~# 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: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:8a:b5:0c brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 192.168.11.38/24 brd 192.168.11.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe8a:b50c/64 scope link
        valid_lft forever preferred_lft forever
root@serverhaproxy2:~# ping 192.168.12.1
PING 192.168.12.1 (192.168.12.1) 56(84) bytes of data.
64 bytes from 192.168.12.1: icmp_seq=1 ttl=63 time=1.15 ms
^C64 bytes from 192.168.12.1: icmp_seq=2 ttl=63 time=1.26 ms

--- 192.168.12.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.148/1.201/1.255/0.053 ms
root@serverhaproxy2:~#
```

La machine a également été référencer sur le serveur DNS et il y'a également un test que ca fonctionne grâce à nslookup :

```

debianclient~# ifconfig
serverhaproxy  IN      A       192.168.11.36
mariadb-2      IN      A       192.168.11.37
serverhaproxy2 IN      A       192.168.11.38

smtp  IN      CNAME   servermail
imap  IN      CNAME   servermail
~
~
~
~/var/cache/bind/db.menuimetal.fr" 45L, 880B écrit(s)
root@dns:~# systemctl restart bind9
root@dns:~# nslookup
> 192.168.11.38
** server can't find 38.11.168.192.in-addr.arpa: NXDOMAIN
> serverhaproxy2
Server:      192.168.12.1
Address:     192.168.12.1#53

Name:   serverhaproxy2.menuimetal.fr
Address: 192.168.11.38
>

```

22. Ajouter le second serveur HaProxy à Nagios

Le second serveur a bien été ajouté à nagios avec les mêmes service que le premier :

serverhaproxy	Charge systeme	UNKNOWN	11-21-2024 16:36:16	0d 1h 33m 56s	4/4	Le seuil d'alerte doit être un nombre à virgule flottante!
	Etat du systeme	OK	11-21-2024 16:38:12	0d 1h 45m 55s	1/4	PING OK - Paquets perdus = 0%, RTA = 1.01 ms
	Load Average	OK	11-21-2024 16:39:32	0d 2h 9m 40s	1/4	OK - Charge moyenne: 0.03, 0.02, 0.00
	Local Disk	OK	11-21-2024 16:35:52	0d 2h 7m 20s	1/4	DISK OK - free space: / 26779 MiB (90,74% inode=97%):
	PING STATUS	OK	11-21-2024 16:35:30	0d 0h 15m 46s	1/4	PING OK - Paquets perdus = 0%, RTA = 1.25 ms
	Taux de remplissage du disque dur	OK	11-21-2024 16:38:22	0d 1h 31m 50s	1/4	DISK OK - free space: / 26779 MiB (90,74% inode=97%):
serverhaproxy2	Charge systeme	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:41:21 CET 2024
	Etat du systeme	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:42:38 CET 2024
	Load Average	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:43:55 CET 2024
	Local Disk	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:44:04 CET 2024
	PING STATUS	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:41:29 CET 2024
	Taux de remplissage du disque dur	PENDING	N/A	0d 0h 0m 8s+	1/4	Service check scheduled for Thu Nov 21 16:42:46 CET 2024

Depuis le serveur nagios j'ai rajouté un fichier de

configuration pour le serveur haproxy2 lui même afin de met ses services comme montré ci-dessus :

```
root@srv-nagios:/usr/local/nagios/etc/servers# ls
debian-client.cfg  omvdebian.cfg  serverdhcpdebian.cf  serverhaproxy2.cfg  serverhaproxy.cfg  serverradius.cfg  serversyslog.cfg  serverweb1.cfg
root@srv-nagios:/usr/local/nagios/etc/servers#
```

23. Installer Heartbeat sur les deux serveurs.

Nous avons installé le paquet heartbeat sur les deux serveur haproxy1 et 2 :

```
root@serverhaproxy:~# dpkg -l | grep heartbeat
ii  heartbeat          1:3.0.6-13+b2      amd64      Subsystem for High-Availability Linux
ii  libheartbeat2      1:3.0.6-13+b2      amd64      Subsystem for High-Availability Linux (libraries)
```

```
root@serverhaproxy2:~# dpkg -l | grep heartbeat
ii  heartbeat          1:3.0.6-13+b2      amd64      Subsystem for High-Availability Linux
ii  libheartbeat2      1:3.0.6-13+b2      amd64      Subsystem for High-Availability Linux (libraries)
root@serverhaproxy2:~#
```

24. Configurer les fichiers ha.cf et haresources.

Il faut ensuite créer les fichier de configuration ha.cf dans **/etc/ha.d/ha.cf** et le fichier haresource :

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

Nous mettons donc ce contenu dans le fichier sur les deux serveurs, voici le contenu et une explication :

```
logfacility local0          # Définit où les logs seront envoyés
keepalive 2                # Fréquence (en secondes) des "messages de vie"
deadtime 10                # Temps avant de déclarer un nœud comme "mort"
udpport 694                # Port utilisé pour la communication entre nœuds
bcast ens18                # Méthode de communication (broadcast sur eth0)
node serverhaproxy         # Nom du premier nœud
node serverhaproxy[2]     # Nom du second nœud
```

Dans le fichier haressource j'ai eu une erreur critique pour mettre en place heartbeat car dans ce fichier de configuration la syntaxe était pas bonne il faut rajouter le masque et l'interface réseau : j'ai simplement mis

serverhaproxy : Nom du nœud principal.

192.168.1.100 : Adresse IP flottante (virtuelle) du

cluster.

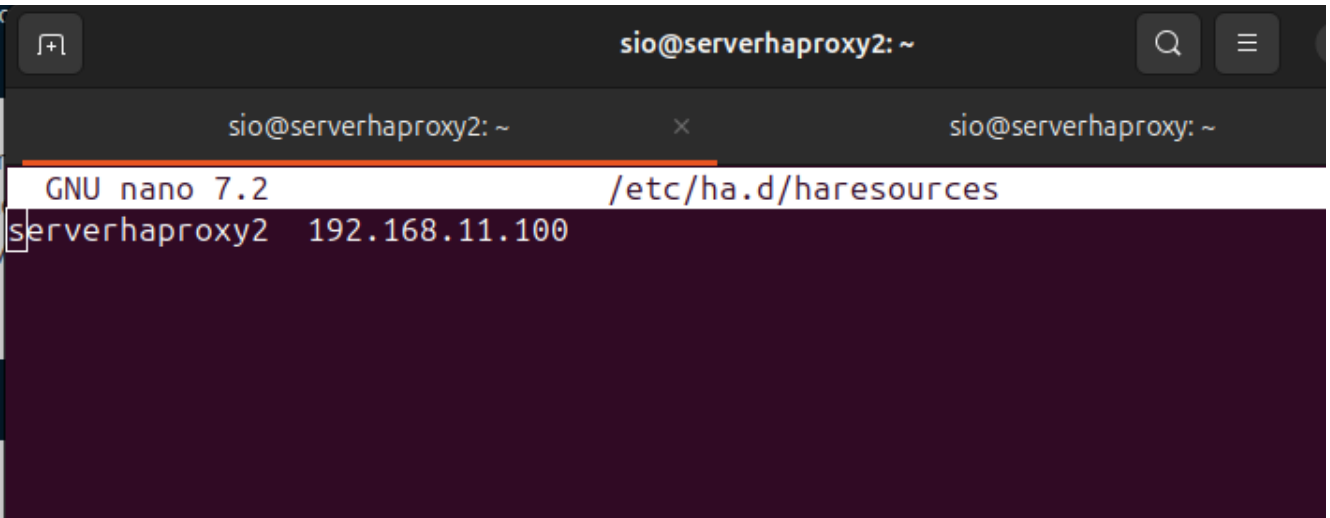
```
serverhaproxy 192.168.11.100
```

Il faut également référencer le nom de domaine et le DNS afin que le service trouve le nom attribué (serverhaproxy) qui est dans le fichier haressource :

```
search menuimetal.fr  
nameserver 192.168.12.1  
#nameserver 172.16.0.4
```

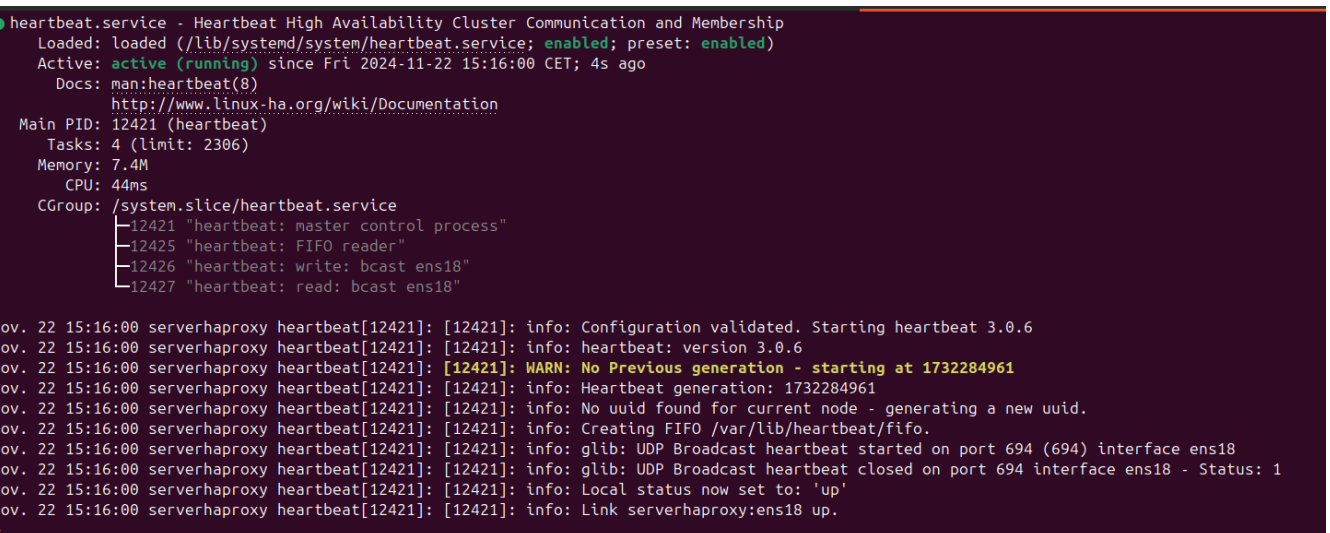
Ici aussi j'avais fais une erreur c'était de faire la même syntax avec un nom différent, l'ip de la machine virtuelle va s'attribuer que sur une machine dont la deuxième, ce n'est pas ce qu'ont veux, ont veux que seul la machine

haproxy 1 à une ip virtuelle :



```
sio@serverhaproxy2: ~  
GNU nano 7.2 /etc/ha.d/haresources  
serverhaproxy2 192.168.11.100
```

Voici les erreurs rencontré :



```
● heartbeat.service - Heartbeat High Availability Cluster Communication and Membership  
Loaded: loaded (/lib/systemd/system/heartbeat.service; enabled; preset: enabled)  
Active: active (running) since Fri 2024-11-22 15:16:00 CET; 4s ago  
Docs: man:heartbeat(8)  
       http://www.linux-ha.org/wiki/Documentation  
Main PID: 12421 (heartbeat)  
Tasks: 4 (limit: 2306)  
Memory: 7.4M  
CPU: 44ms  
CGroup: /system.slice/heartbeat.service  
├─12421 "heartbeat: master control process"  
├─12425 "heartbeat: FIFO reader"  
├─12426 "heartbeat: write: bcast ens18"  
└─12427 "heartbeat: read: bcast ens18"  
  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: Configuration validated. Starting heartbeat 3.0.6  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: heartbeat: version 3.0.6  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: WARN: No Previous generation - starting at 1732284961  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: Heartbeat generation: 1732284961  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: No uuid found for current node - generating a new uuid.  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: Creating FIFO /var/lib/heartbeat/fifo.  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: glib: UDP Broadcast heartbeat started on port 694 (694) interface ens18  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: glib: UDP Broadcast heartbeat closed on port 694 interface ens18 - Status: 1  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: Local status now set to: 'up'  
Nov. 22 15:16:00 serverhaproxy heartbeat[12421]: [12421]: info: Link serverhaproxy:ens18 up.
```

Par exemple l'interface n'a pas été trouvé, ou le fichier de clé n'avais pas la bonne sécurité des droits :

```
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: heartbeat: udpport setting must precede media statementsNov 22 14:40:32 serverhaproxy heartbeat: [12111]:
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: [12111]: ERROR: glib: Get broadcast for interface eth0 failed: No such device
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: Nov 22 14:40:32 serverhaproxy heartbeat: [12111]: ERROR: glib: IP interface [eth0] does not exist
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: [12111]: ERROR: glib: IP interface [eth0] does not exist
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: Nov 22 14:40:32 serverhaproxy heartbeat: [12111]: ERROR: Illegal bcast [UDP/IP broadcast] in config file
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: [12111]: ERROR: Illegal bcast [UDP/IP broadcast] in config file [eth0]
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: Nov 22 14:40:32 serverhaproxy heartbeat: [12111]: ERROR: Heartbeat not started configuration error.
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: [12111]: ERROR: Heartbeat not started: configuration error.
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: Nov 22 14:40:32 serverhaproxy heartbeat: [12111]: ERROR: Configuration error, heartbeat not started.
nov. 22 14:40:32 serverhaproxy heartbeat[12111]: [12111]: ERROR: Configuration error, heartbeat not started.
nov. 22 14:40:32 serverhaproxy systemd[1]: heartbeat.service: Main process exited, code=exited, status=6/NOTCONFIGURED
nov. 22 14:40:32 serverhaproxy systemd[1]: heartbeat.service: Failed with result 'exit-code'.
nov. 22 14:45:57 serverhaproxy systemd[1]: Started heartbeat.service - Heartbeat High Availability Cluster Communication and Membership.
nov. 22 14:45:57 serverhaproxy heartbeat[12127]: heartbeat: udpport setting must precede media statementsNov 22 14:45:57 serverhaproxy heartbeat: [12127]:
nov. 22 14:45:57 serverhaproxy heartbeat[12127]: [12127]: ERROR: Cannot open keyfile [/etc/ha.d/authkeys]. Stop.
nov. 22 14:45:57 serverhaproxy heartbeat[12127]: Nov 22 14:45:57 serverhaproxy heartbeat: [12127]: ERROR: Authentication configuration error.
nov. 22 14:45:57 serverhaproxy heartbeat[12127]: [12127]: ERROR: Authentication configuration error.
nov. 22 14:45:57 serverhaproxy heartbeat[12127]: Nov 22 14:45:57 serverhaproxy heartbeat: [12127]: ERROR: Configuration error, heartbeat not started.
nov. 22 14:45:57 serverhaproxy heartbeat[12127]: [12127]: ERROR: Configuration error, heartbeat not started.
nov. 22 14:45:57 serverhaproxy systemd[1]: heartbeat.service: Main process exited, code=exited, status=6/NOTCONFIGURED
nov. 22 14:45:57 serverhaproxy systemd[1]: heartbeat.service: Failed with result 'exit-code'.
nov. 22 14:47:52 serverhaproxy systemd[1]: Started heartbeat.service - Heartbeat High Availability Cluster Communication and Membership.
nov. 22 14:47:52 serverhaproxy heartbeat[12136]: heartbeat: udpport setting must precede media statementsNov 22 14:47:52 serverhaproxy heartbeat: [12136]:
nov. 22 14:47:52 serverhaproxy heartbeat[12136]: [12136]: ERROR: Bad permissions on keyfile [/etc/ha.d/authkeys], 600 recommended.
nov. 22 14:47:52 serverhaproxy heartbeat[12136]: Nov 22 14:47:52 serverhaproxy heartbeat: [12136]: ERROR: Authentication configuration error.
lines 1-42]
```

Voici une correction des erreurs :

```
serverhaproxy 192.168.11.100/24/ens18
```

```
root@serverhaproxy2:/etc/ha.d# systemctl status heartbeat.service
● heartbeat.service - Heartbeat High Availability Cluster Communication and Membership
   Loaded: loaded (/lib/systemd/system/heartbeat.service; enabled; preset: enabled)
   Active: active (running) since Fri 2024-11-22 15:53:24 CET; 8s ago
     Docs: man:heartbeat(8)
           http://www.linux-ha.org/wiki/Documentation
  Main PID: 6020 (heartbeat)
    Tasks: 4 (limit: 2306)
   Memory: 7.0M
      CPU: 62ms
   CGroup: /system.slice/heartbeat.service
           └─6020 "heartbeat: master control process"
             └─6024 "heartbeat: FIFO reader"
               └─6025 "heartbeat: write: bcast ens18"
                 └─6026 "heartbeat: read: bcast ens18"

nov. 22 15:53:24 serverhaproxy2 heartbeat[6020]: [6020]: info: glib: UDP Broadcast heartbeat started on port 694 (694) interface ens18
nov. 22 15:53:24 serverhaproxy2 heartbeat[6020]: [6020]: info: glib: UDP Broadcast heartbeat closed on port 694 interface ens18 - Status: 1
nov. 22 15:53:24 serverhaproxy2 heartbeat[6020]: [6020]: info: Local status now set to: 'up'
nov. 22 15:53:24 serverhaproxy2 heartbeat[6020]: [6020]: info: Link serverhaproxy2:ens18 up.
nov. 22 15:53:26 serverhaproxy2 heartbeat[6020]: [6020]: info: Link serverhaproxy:ens18 up.
nov. 22 15:53:26 serverhaproxy2 heartbeat[6020]: [6020]: info: Status update for node serverhaproxy: status active
nov. 22 15:53:26 serverhaproxy2 heartbeat[6028]: [6028]: debug: notify_world: setting SIGCHLD Handler to SIG_DFL
nov. 22 15:53:26 serverhaproxy2 harc(default)[6035]: info: Running /etc/ha.d/rc.d/status status
nov. 22 15:53:26 serverhaproxy2 heartbeat[6020]: [6020]: info: Comm_now_up(): updating status to active
nov. 22 15:53:26 serverhaproxy2 heartbeat[6020]: [6020]: info: Local status now set to: 'active'
```

On peut constater que une ip virtuelle à bien été ajouté sur notre machine linux haproxy et que la communication fonctionne :

```
root@serverhaproxy:/etc/ha.d# 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: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:c6:ac:8c brd ff:ff:ff:ff:ff:ff
    altname_ens18
    inet 192.168.11.36/24 brd 192.168.11.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet 192.168.11.100/24 brd 192.168.11.255 scope global secondary ens18:0
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe0a:ac8c/64 scope link
        valid_lft forever preferred_lft forever
root@serverhaproxy:/etc/ha.d#
```

```
root@serverhaproxy:/etc/ha.d# ping 192.168.11.100
PING 192.168.11.100 (192.168.11.100) 56(84) bytes of data.
64 bytes from 192.168.11.100: icmp_seq=1 ttl=64 time=0.147 ms
64 bytes from 192.168.11.100: icmp_seq=2 ttl=64 time=0.060 ms
^C
--- 192.168.11.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1025ms
rtt min/avg/max/mdev = 0.060/0.103/0.147/0.043 ms
root@serverhaproxy:/etc/ha.d#
```

Si nous mettons l'adresse ip virtuelle sur un navigateur le service web fonctionne correctement :



25. Tester le basculement en arrêtant un serveur HaProxy.

Nous allons d'abord arrêter un serveur web afin de voir l'efficacité de notre haproxy et l'ip virtuelle :

```
root@web-1:~# systemctl stop apache2  
root@web-1:~#
```

Le haproxy nous dit bien que le serveur 1 est down :

```
root@serverhaproxy:/etc/ha.d# systemctl status haproxy.service
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-11-21 16:00:32 CET; 23h ago
     Docs: man:haproxy(1)
          file:/usr/share/doc/haproxy/configuration.txt.gz
   Main PID: 9373 (haproxy)
     Tasks: 2 (limit: 2306)
    Memory: 41.7M
       CPU: 30.189s
   CGroup: /system.slice/haproxy.service
           └─9373 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
           └─9375 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

nov. 22 15:58:45 serverhaproxy haproxy[9375]: Server backend_webservers/web1 is DOWN, reason: Layer4 connection problem, info: "Connection refused", check duration: 0ms. 2 active and 0 back
nov. 22 15:58:46 serverhaproxy haproxy[9375]: 192.168.11.251:37338 [22/Nov/2024:15:58:46.059] front_web_servers backend_webservers/web3 0/0/2/1/3 200 1106 - - - - - 2/1/0/0/0 0/0 "GET / HTTP
```

Sa fonctionne correctement :



Nous allons maintenant stopper le service de haproxy 1 pour voir si le second haproxy 2 prend le relais :

```
root@serverhaproxy:/etc/ha.d# systemctl status haproxy.service
○ haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
   Active: inactive (dead) since Fri 2024-11-22 16:04:17 CET; 3s ago
     Duration: 1d 3min 45.074s
     Docs: man:haproxy(1)
           file:/usr/share/doc/haproxy/configuration.txt.gz
   Process: 9373 ExecStart=/usr/sbin/haproxy -Ws -f $CONFIG -p $PIDFILE $EXTRA_OPTS (code=exited, status=0/SUCCESS)
  Main PID: 9373 (code=exited, status=0/SUCCESS)
    CPU: 30.314s

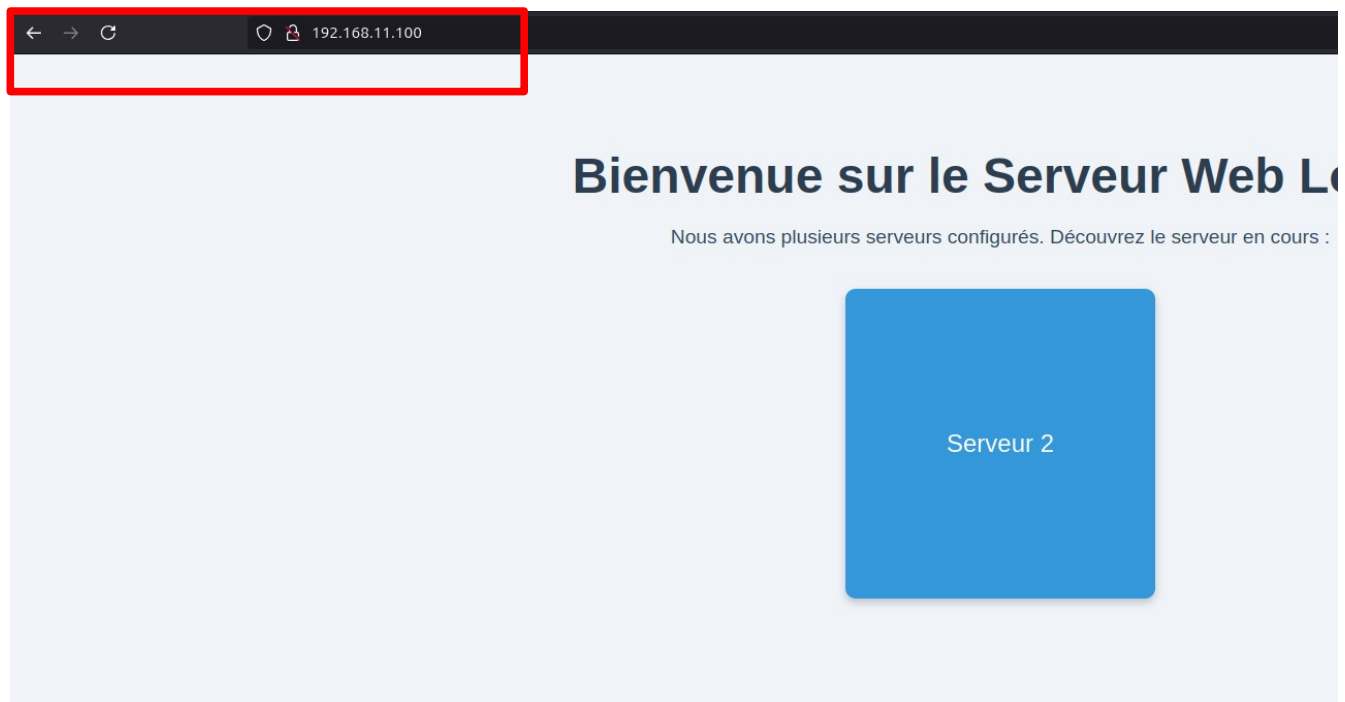
nov. 22 16:04:16 serverhaproxy haproxy[9375]: 192.168.11.35:52460 [22/Nov/2024:16:04:16.155] front_stats front_stats/<STATS> 0/0/0/0
nov. 22 16:04:17 serverhaproxy haproxy[9373]: [NOTICE] (9373) : haproxy version is 2.6.12-1+deb12u1
nov. 22 16:04:17 serverhaproxy haproxy[9373]: [NOTICE] (9373) : path to executable is /usr/sbin/haproxy
nov. 22 16:04:17 serverhaproxy haproxy[9373]: [WARNING] (9373) : Exiting Master process...
nov. 22 16:04:17 serverhaproxy systemd[1]: Stopping haproxy.service - HAProxy Load Balancer...
nov. 22 16:04:17 serverhaproxy haproxy[9373]: [ALERT] (9373) : Current worker (9375) exited with code 143 (Terminated)
nov. 22 16:04:17 serverhaproxy haproxy[9373]: [WARNING] (9373) : All workers exited. Exiting... (0)
nov. 22 16:04:17 serverhaproxy systemd[1]: haproxy.service: Deactivated successfully.
nov. 22 16:04:17 serverhaproxy systemd[1]: Stopped haproxy.service - HAProxy Load Balancer.
nov. 22 16:04:17 serverhaproxy systemd[1]: haproxy.service: Consumed 30.314s CPU time.
root@serverhaproxy:/etc/ha.d#
```

Si vous êtes attentif on peut voir que le serveur 2 prend en compte les serveurs web down et up :

```
root@serverhaproxy2:/etc/ha.d# systemctl status haproxy.service
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; preset: enabled)
   Active: active (running) since Thu 2024-11-21 16:21:39 CET; 23h ago
     Docs: man:haproxy(1)
           file:/usr/share/doc/haproxy/configuration.txt.gz
  Main PID: 456 (haproxy)
    Tasks: 2 (limit: 2306)
   Memory: 49.7M
      CPU: 24.848s
   CGroup: /system.slice/haproxy.service
           └─456 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock
             └─459 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy-master.sock

nov. 21 16:21:38 serverhaproxy2 systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
nov. 21 16:21:39 serverhaproxy2 haproxy[456]: [NOTICE] (456) : New worker (459) forked
nov. 21 16:21:39 serverhaproxy2 haproxy[456]: [NOTICE] (456) : Loading success.
nov. 21 16:21:39 serverhaproxy2 systemd[1]: Started haproxy.service - HAProxy Load Balancer.
nov. 22 15:58:45 serverhaproxy2 haproxy[459]: [WARNING] (459) : Server backend_websevers/web1 is DOWN, reason: Layer4 connecti
nov. 22 15:58:45 serverhaproxy2 haproxy[459]: Server backend_websevers/web1 is DOWN, reason: Layer4 connection problem, info: "
nov. 22 15:58:45 serverhaproxy2 haproxy[459]: Server backend_websevers/web1 is DOWN, reason: Layer4 connection problem, info: "
nov. 22 16:00:57 serverhaproxy2 haproxy[459]: [WARNING] (459) : Server backend_websevers/web1 is UP, reason: Layer4 check pass
nov. 22 16:00:57 serverhaproxy2 haproxy[459]: Server backend_websevers/web1 is UP, reason: Layer4 check passed, check duration:
nov. 22 16:00:57 serverhaproxy2 haproxy[459]: Server backend_websevers/web1 is UP, reason: Layer4 check passed, check duration:
root@serverhaproxy2:/etc/ha.d#
```

Ca fonctionne :



26. Remonter les logs Heartbeat sur rsyslog.

Pour la première étape il faut vérifier que rsyslog est bien installé sur ma machine qui doit envoyer les logs donc il faut regarder le statut du service rsyslog :

Il faut ensuite configurer le fichier `/etc/rsyslog.conf` de ma machine haproxy en ajoutant la ligne `*.*@@192.168.11.31:514` qui va permettre de envoyer les logs vers le serveur central (serversyslog)

Depuis notre serveur rsyslog, nous pouvons voir les logs de haproxy, par exemple le service heartbeat stoppé ou la demande d'une ip virtuelle pour notre serveur haproxy ou l'étude de notre interface réseau pour l'ip virtuelle :

```
2024-11-22T15:53:22+01:00 serverhaproxy2 /usr/lib/ocf/resource.d/heartbe INFO: Resource is stopped
2024-11-22T15:53:22+01:00 serverhaproxy heartbeat: [15352]: debug: notify_world: setting SIGCHLD Handler to SIG_DFL
2024-11-22T15:53:22+01:00 serverhaproxy harc(default)[15352]: info: Run
2024-11-22T15:53:22+01:00 serverhaproxy ip-request-resp(default)[15352]: received ip-request-resp 192.168.11.100/24/ens18 OK no
2024-11-22T15:53:22+01:00 serverhaproxy ResourceManager(default)[15369]: info: Acquiring resource group: serverhaproxy 192.168.11.100/24/ens18
2024-11-22T15:53:22+01:00 serverhaproxy /usr/lib/ocf/resource.d/heartbe INFO: Resource is stopped
2024-11-22T15:53:22+01:00 serverhaproxy ResourceManager(default)[15369]: info: Running /etc/ha.d/resource.d/IPaddr 192.168.11.100/24/ens18 start
2024-11-22T15:53:22+01:00 serverhaproxy IPaddr(IPaddr_192.168.11.100)[15369]: INFO: Using calculated netmask for 192.168.11.100: 255.255.255.0
2024-11-22T15:53:23+01:00 serverhaproxy IPaddr(IPaddr_192.168.11.100)[15369]: INFO: eval ifconfig ens18:0 192.168.11.100 netmask 255.255.255.0 broadcast 192.168.11.255
2024-11-22T15:53:23+01:00 serverhaproxy /usr/lib/ocf/resource.d/heartbe INFO: Success
```

```
2024-11-22T16:04:17+01:00 serverhaproxy systemd[1]: Stopping haproxy.service - HAProxy Load Balancer...
2024-11-22T16:04:17+01:00 serverhaproxy systemd[1]: haproxy.service: Deactivated successfully.
2024-11-22T16:04:17+01:00 serverhaproxy systemd[1]: Stopped haproxy.service - HAProxy Load Balancer.
2024-11-22T16:04:17+01:00 serverhaproxy systemd[1]: haproxy.service: Consumed 30.314s CPU time.
```

