Mattéo Sami Nilda Maxime BTS SIO 1 Semestre 2





# <u>CAS 1 -</u> <u>Entreprise RECTILINE</u>

## <u>Sommaire</u>

Lot 1 - Migration du serveur de base de données (Maxime et Nilda)	3
Lot 2 – Mise en place du nouveau plan d'adressage et segmentation du réseau (Sami et Mattéo)	6
Lot 3 – Implantation du nouveau serveur de base de données (Maxime, Nilda, Mattéo et	
Sami)	10
Diagramme des tâches	13

#### Lot 1 - Migration du serveur de base de données (Maxime et Nilda)

Tout d'abord, nous allons procéder à une mise à jour du système (avant de procéder à la mise à jour du système, vérifier si le fichier sources.list contient les bonnes informations en faisant nano /etc/apt/sources.list dans le terminal Linux) avec la commande suivante :

root@deb1-b319:/home/administrateur# apt update Atteint :1 https://ftp.debian.org/debian bookworm I nRelease Réception de :2 https://ftp.debian.org/debian bookw orm-updates InRelease [52,1 kB] Réception de :3 https://ftp.debian.org/debian bookw orm-proposed-updates InRelease [59,5 kB] Réception de :4 https://ftp.debian.org/debian bookw

Apt update : commande qui permet de mettre à jour le référentiel APT(Advanced Persistent Threat) de votre appareil. Elle est à utiliser lorsqu'on modifie, ajoute une source ou pour récupérer les dernières modifications des dépôts.

Pour installer le serveur de bases de données MariaDB sous Debian 12, nous allons utiliser la commande suivante :

root@deb1-b319:/home/administrateur# apt-get instal l mariadb-server Lecture des listes de paquets... Fait Construction de l'arbre des dépendances... Fait Lecture des informations d'état... Fait Les paquets supplémentaires suivants seront install és : galera-4 gawk libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl libdaxctl1 libdbd-mariadb-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-template-perl libmariadb3 libndctl6 libpmem1 libsigsegv2 libterm-readkey-perl liburing2 mariadb-client

MariaDB est un serveur de bases de données relationnelles Open Source.

Un serveur de bases de données permet de **stocker les données dans des tables séparées** afin d'organiser correctement la base de données.

 Après avoir installé le serveur de base de données MariaDB sur la machine serveur, nous pouvons avoir accès à la BDD en local avec la commande :

```
root@deb1-b319:/home/administrateur# mariadb -u roo
t -p
Enter password:
Welcome to the MariaDB monitor. Commands end with
; or \g.
Your MariaDB connection id is 31
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12
Copyright (c) 2000, 2018, Oracle, MariaDB Corporati
on Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear t
he current input statement.
```

- Nous allons créer la base de données en insérant des données dans celle-ci et en attribuant des accès à la base de données :
  - 1. Création de la base de données :



2. Direction vers la BDD, création de la table "test" et insertion de données :



3. Affichage de la table "test" et attribution des accès à la BDD :

ariaDB [testserv]> SELECT * FROM test;
id   libelleTest
A001   Première donnée
A002   Deuxième donnée
A003   Troisième donnée
rows in set (0,000 sec)
ariaDB [testserv]> GRANT ALL PRIVILEGES ON testserv.* TO admin@'%' IDENTIFIED B
'admin';
uery OK, 0 rows affected (0,006 sec)
ariaDB [testserv]> GRANT ALL PRIVILEGES ON testserv.* TO admin@'localhost' IDEN
IFIED BY 'admin';
uery OK, 0 rows affected (0,004 sec)

- Test de l'interconnexion d'une des machines client à la machine serveur :

```
root@Debian-12-Bookworm:~# ping 172.16.26.5
PING 172.16.26.5 (172.16.26.5) 56(84) bytes of data.
64 bytes from 172.16.26.5: icmp_seq=1 ttl=64 time=1.35 ms
64 bytes from 172.16.26.5: icmp_seq=2 ttl=64 time=1.45 ms
64 bytes from 172.16.26.5: icmp_seq=3 ttl=64 time=1.53 ms
64 bytes from 172.16.26.5: icmp_seq=4 ttl=64 time=1.48 ms
64 bytes from 172.16.26.5: icmp_seq=5 ttl=64 time=1.60 ms
^C
--- 172.16.26.5 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 1.348/1.480/1.602/0.084 ms
```

On remarque que la connexion entre les deux machines est fonctionnelle !

- Connexion à distance sur le serveur de BDD avec la commande suivante :

```
root@Debian-12-Bookworm:~# mariadb -u admin -p -h 172.16.26.5
Enter password:
welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12
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)]> use testserv
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

On a réussi à se connecter au serveur de BDD sur la machine client !

```
MariaDB [testserv]> SELECT * FROM test;
+----+
| id | libelleTest |
+----+
| A001 | Première donnée |
| A002 | Deuxième donnée |
| A003 | Troisième donnée |
+----+
3 rows in set (0,002 sec)
```

On a réussi à afficher la table test sur la machine client !

### Lot 2 – Mise en place du nouveau plan d'adressage et segmentation du réseau (Sami et Mattéo)

Nous pouvons voir que dans cette partie nous devons prévoir un plan d'adressages avec trois sous réseaux les voici :

Les numéros des réseaux et le masque de sous-réseau :

192.168.111.2 /24 : (Sami) 192.168.111.1 /24 : (Mattéo)

Configurer les stations, tester et valider la mise en place du nouveau réseau.

- Documenter les installations, les configurations, les contrôles effectués.

Le schéma logique du réseau avec les adresses IP affectés :



Détailler les trois sous-réseaux et les affectations des adresses IP :

```
interface Vlan1
no ip address
!
interface Vlan10
ip address 192.168.111.1 255.255.255.192
!
interface Vlan20
ip address 192.168.111.65 255.255.255.192
!
interface Vlan30
ip address 192.168.111.129 255.255.255.192
```

Configurer les commutateurs ; créer, nommer et configurer les VLAN, affecter les ports.

Pour le VLAN 10 nous avons :

Sami qui a une adresse ip de 192.168.111.2 et Matteo qui a une adresse 192.168.111.1. Ici nous pouvons voir que le commutateur 1 ping le commutateur 2 et voici le résultat ci-dessous :

root@debian:/home/administrateur# ping 192.168.111.2
PING 192.168.111.2 (192.168.111.2) 56(84) bytes of data.
64 bytes from 192.168.111.2: icmp seq=1 ttl=64 time=0.964 ms
64 bytes from 192.168.111.2: icmp seg=2 ttl=64 time=1.09 ms
64 bytes from 192.168.111.2: icmp seg=3 ttl=64 time=1.12 ms
64 bytes from 192,168,111.2; icmp_seg=4 ttl=64 time=1.13 ms
64 bytes from 192.168.111.2: icmp_seg=5 ttl=64 time=1.19 ms
^C
192.168.111.2 ping statistics
5 packets transmitted, 5 received, 0% packet loss, time 4031ms
rtt min/avg/max/mdev = 0.964/1.103/1.195/0.077 ms
root@debian:/home/administrateur# ning 192 168 111 2
DING 102 168 111 2 (102 168 111 2) 56(84) bytes of data
^r
102 160 111 2 pipe statistics
<pre> 192.100.111.2 piny Statistics 6 packets transmitted 0 received 100% packet loss time EllOme.</pre>
o packets transmitted, o received, 100% packet toss, time siloms
restadebier, (here ladministrateur)
LOO (GOGDTAU: / HOINE/ AGNITUT2 (LG (GAL#

Pour le VLAN 20 nous avons :

Sami qui a une adresse ip de 192.168.111.67 et Matteo qui a une adresse ip de 192.168.111.80. Ici nous pouvons voir que le commutateur 1 ping le commutateur 2 et voici le résultat ci-dessous :

```
root@deb1-b31905:/home/administrateur# ping 192.168.111.67
PING 192.168.111.67 (192.168.111.67) 56(84) bytes of data.
64 bytes from 192.168.111.67: icmp_seq=1 ttl=64 time=0.925 ms
64 bytes from 192.168.111.67: icmp_seq=2 ttl=64 time=0.963 ms
64 bytes from 192.168.111.67: icmp_seq=3 ttl=64 time=0.991 ms
64 bytes from 192.168.111.67: icmp_seq=4 ttl=64 time=0.956 ms
^C
--- 192.168.111.67 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3038ms
rtt min/avg/max/mdev = 0.925/0.958/0.991/0.044 ms
```

Ici nous pouvons voir que le commutateur 2 ping le commutateur 1 et voici le résultat ci-dessous :

Pour le VLAN 30 nous avons :

Sami qui a une adresse ip de 192.168.111.135 et Matteo qui a une adresse 192.168.111.131 Ici nous pouvons voir que le commutateur 2 ping le commutateur 1 et voici le résultat

ci-dessous

root@deblan:/home# ping 192.168.111.131
PING 192.168.111.131 (192.168.111.131) 56(84) bytes of data.
64 bytes from 192.168.111.131: icmp\_seq=1 ttl=64 time=1.72 ms
64 bytes from 192.168.111.131: icmp\_seq=2 ttl=64 time=0.914 ms
64 bytes from 192.168.111.131: icmp\_seq=3 ttl=64 time=0.984 ms
64 bytes from 192.168.111.131: icmp\_seq=4 ttl=64 time=0.891 ms
^C
--- 192.168.111.131 ping statistics --4 packets transmitted, 4 received, 0% packet loss, time 3013ms
rtt min/avg/max/mdev = 0.891/1.128/1.724/0.346 ms

Ici nous pouvons voir que le commutateur 1 ping le commutateur 2 et voici le résultat ci-dessous :

root@deb1-b31905:/home# ping 192.168.111.135
PING 192.168.111.135 (192.168.111.135) 56(84) bytes of data.
64 bytes from 192.168.111.135: icmp\_seq=1 ttl=64 time=0.639 ms
64 bytes from 192.168.111.135: icmp\_seq=2 ttl=64 time=0.927 ms
64 bytes from 192.168.111.135: icmp\_seq=3 ttl=64 time=0.907 ms
64 bytes from 192.168.111.135: icmp\_seq=4 ttl=64 time=0.907 ms
7C
--- 192.168.111.135 ping statistics --4 packets transmitted, 4 received, 0% packet loss, time 3024ms
rtt min/avg/max/mdev = 0\_639/0.842/0.927/0.121 ms

Nous pouvons constater que si prenons un vlan 10 pour communiquer à un vlan 30 cela ne fonctionnera il nous montrera que la destination est impossible car il ne font pas partie du même vlan alors que si le commutateur 1 et le commutateur 2 ont le même vlan ils peuvent se communiquer entre eux sans problème.

```
root@deb1-b31905:/home# ping 192.168.111.135
PING 192.168.111.135 (192.168.111.135) 56(84) bytes of data.
From 192.168.111.125 icmp_seq=1 Destination Host Unreachable
From 192.168.111.125 icmp_seq=2 Destination Host Unreachable
From 192.168.111.125 icmp_seq=3 Destination Host Unreachable
^C
--- 192.168.111.135 ping statistics ---
6 packets transmitted, 0 received, +3 errors, 100% packet loss, time 5295ms
pipe 4
```

Documenter les installations, les configurations, les contrôles effectués.

Ici, nous pouvons observer que l'interface du vlan 10 contient 0/8 Fastethernet.

interface FastEthernet0/1 switchport access vlan 10
! interface FastEthernet0/2 switchport access vlan 10
: interface FastEthernet0/3 switchport access vlan 10
interface FastEthernet0/4 switchport access vlan 10
interface FastEthernet0/5 switchport access vlan 10
interface FastEthernet0/6 switchport access vlan 10
interface FastEthernet0/7 switchport access vlan 10
interface FastEthernet0/8 switchport access vlan 10

Ici, nous pouvons observer que l'interface du vlan 20 s'ajoute à partir du 0/9 jusqu'à le 0/16 Fastethernet.



Ici, nous pouvons observer que l'interface du vlan 30 s'ajoute à partir du 0/16 jusqu'à le 0/24 Fastethernet. Nous pouvons observer que tous les vlans ont un fastethernet qui les séparent au fur et à mesure et ainsi ils contiennent 0/2 Gigabit ethernet.

Un vlan (Virtual local area network): Il s'agit, sur un même switch, de créer plusieurs réseaux indépendants ne pouvant pas, par défaut, communiquer entre eux.

Gigabit Ethernet est une **technologie de transmission qui s'appuie sur le format de trame et protocole Ethernet, utilisé dans les réseaux locaux** (LAN, Local Area Network). Elle fournit un débit de données d'un million de bits par seconde (soit un gigabit).

Nous pouvons aussi observer que les ports trunk sont les trames reçues sur une interface qui sont supposées avoir des balises VLAN. Ainsi, les ports agrégés sont destinés aux liaisons entre des commutateurs ou d'autres périphériques réseau et peuvent transporter du trafic pour plusieurs VLAN.





Installation et activation du package vlan avec modprobe 8021q :

- On va sur le fichier /etc/network/interfaces (en faisant nano /etc/network/interfaces)
- se mettre en dhcp
- réactiver les cartes avec /etc/init.d/networking restart
- installation du package vlan (apt install vlan)
- activation de ce package avec modprobe 8021q

Ouvrir votre fichier d'interface, et remettre en DHCP pour pouvoir télécharger le paquet VLAN :

# nano /etc/network/interfaces

auto enp0s3 iface enp0s3 inet dhcp

Réactiver les cartes :

# /etc/init.d/networking restart

Installer le package vlan, et l'activer :

# apt install vlan # modprobe 8021q

#### Adresses virtuelles mises en places pour la machine serveur (VLAN 10, VLAN 20, VLAN 30) :

#iface enp0s3.10 inet static
#vlan-raw-device enp0s3
#address 192.168.111.1
#netmask 255.255.255.192
#iface enp0s3.20 inet static
#vlan-raw-device enp0s3
#address 192.168.111.65
#netmask 255.255.255.192
#iface enp0s3.30 inet static
#vlan-raw-device enp0s3
#address 192.168.111.130
#netmask 255.255.255.192

Il faut bien penser à réactiver les vlans avec sudo ifdown enp0s3.(numéro du vlan) et sudo ifup enp0s3.(numéro du vlan).

Vérification de la présence des vlans :

root@deb1-b319:/home/administrateur# cat /proc/net/vlan/config
VLAN Dev name   VLAN ID
Name-Type: VLAN_NAME_TYPE_RAW_PLUS_VID_NO_PAD
enp0s3.10   10   enp0s3
enp0s3.20   20   enp0s3
enp0s3.30   30   enp0s3
root@deb1-b319:/home/administrateur#

Les 3 VLANS sont bien présents !

Tests de vérification :

1. Test 1 pour vérifier la configuration du vlan 10 (en mettant tout dans le même vlan) :

1	root@SamiBDebian-12B316:/home/administrateur# ping 192.168.111.2
F	PING 192.168.111.2 (192.168.111.2) 56(84) bytes of data.
é	64 bytes from 192.168.111.2: icmp_seq=2 ttl=255 time=0.899 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=3 ttl=255 time=0.884 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=4 ttl=255 time=0.886 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=5 ttl=255 time=0.858 ms
1	64 bytes from 192.168.111.2: icmp_seq=6 ttl=255 time=0.845 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=7 ttl=255 time=0.853 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=8 ttl=255 time=0.866 ms
ŝ	54 bytes from 192.168.111.2: icmp_seq=9 ttl=255 time=2.02 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=10 ttl=255 time=0.840 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=11 ttl=255 time=0.869 ms
ŝ	64 bytes from 192.168.111.2: icmp_seq=12 ttl=255 time=0.885 ms
ĺ	64 bytes from 192.168.111.2: icmp_seq=13 ttl=255 time=3.60 ms
1	^X64 bytes from 192.168.111.2: icmp_seq=14 ttl=255 time=0.750 ms
ľ	^C
	192.168.111.2 ping statistics
	14 packets transmitted, 13 received, 7.14286% packet loss, time 13152ms
į	rtt min/avg/max/mdev = 0.750/1.158/3.602/0.771 ms

root@Debian-12-Bookworm:~# ping 192.168.111.5
PING 192.168.111.5 (192.168.111.5) 56(84) bytes of data.
64 bytes from 192.168.111.5: icmp\_seq=6 ttl=64 time=0.933 ms

2. <u>Test 2 pour vérifier la configuration du vlan 10 (en mettant la machine client dans un autre vlan (vlan 20))</u>:

```
root@Debian-12-Bookworm:~# ping 192.168.111.5
PING 192.168.111.5 (192.168.111.5) 56(84) bytes of data.
From 192.168.111.2 icmp_seq=1 Destination Host Unreachable
From 192.168.111.2 icmp_seq=2 Destination Host Unreachable
From 192.168.111.2 icmp_seq=3 Destination Host Unreachable
From 192.168.111.2 icmp_seq=4 Destination Host Unreachable
From 192.168.111.2 icmp_seq=5 Destination Host Unreachable
From 192.168.111.2 icmp_seq=6 Destination Host Unreachable
From 192.168.111.2 icmp_seq=7 Destination Host Unreachable
From 192.168.111.2 icmp_seq=8 Destination Host Unreachable
From 192.168.111.2 icmp_seq=9 Destination Host Unreachable
From 192.168.111.2 icmp_seq=10 Destination Host Unreachable
From 192.168.111.2 icmp_seq=11 Destination Host Unreachable
From 192.168.111.2 icmp_seq=12 Destination Host Unreachable
^c
--- 192.168.111.5 ping statistics ---
15 packets transmitted, 0 received, +12 errors, 100% packet loss, time 14090ms
pipe 4
```

3. Test du vlan 10 vers la bdd :

root@SamiBDebian-12B316:/home/administrateur# ping 192.168.111.1	
PING 192.168.111.1 (192.168.111.1) 56(84) bytes of data.	
From 192.168.111.5 icmp_seq=148 Destination Host Unreachable	
From 192.168.111.5 icmp_seq=149 Destination Host Unreachable	
From 192.168.111.5 icmp_seq=150 Destination Host Unreachable	
From 192.168.111.5 icmp_seq=151 Destination Host Unreachable	
From 192.168.111.5 icmp_seq=152 Destination Host Unreachable	
From 192.168.111.5 icmp_seq=153 Destination Host Unreachable	
64 bytes from 192.168.111.1: icmp_seq=154 ttl=64 time=1026 ms	
64 bytes from 192.168.111.1: icmp_seq=155 ttl=64 time=1.89 ms	
64 bytes from 192.168.111.1: icmp_seq=156 ttl=64 time=1.13 ms	
64 bytes from 192.168.111.1: icmp_seq=157 ttl=64 time=1.13 ms	
64 bytes from 192.168.111.1: icmp_seq=158 ttl=64 time=0.921 ms	
64 bytes from 192.168.111.1: icmp_seq=159 ttl=64 time=1.14 ms	
64 bytes from 192.168.111.1: icmp_seq=160 ttl=64 time=1.23 ms	
64 bytes from 192.168.111.1: icmp_seq=161 ttl=64 time=0.998 ms	
64 butos from 100 160 111 1, isro sos-160 tt]-64 time-1 05 ms	

4. Test du vlan 20 vers la BDD :

root@SamiBDebian-12B316:/home/administrateur# ping 192.168.111.65
PING 192.168.111.65 (192.168.111.65) 56(84) bytes of data.
64 bytes from 192.168.111.65: icmp_seq=1 ttl=64 time=1.95 ms
64 bytes from 192.168.111.65: icmp_seq=2 ttl=64 time=1.01 ms
64 bytes from 192.168.111.65: icmp_seq=3 ttl=64 time=1.08 ms
64 bytes from 192.168.111.65: icmp_seq=4 ttl=64 time=1.20 ms
64 bytes from 192.168.111.65: icmp_seq=5 ttl=64 time=0.911 ms
64 bytes from 192.168.111.65: icmp_seq=6 ttl=64 time=1.11 ms
64 bytes from 192.168.111.65: icmp_seq=7 ttl=64 time=1.12 ms
64 bytes from 192.168.111.65: icmp_seq=8 ttl=64 time=0.958 ms
64 bytes from 192.168.111.65: icmp_seq=9 ttl=64 time=1.07 ms
64 bytes from 192.168.111.65: icmp_seq=10 ttl=64 time=1.20 ms
64 bytes from 192.168.111.65: icmp_seq=11 ttl=64 time=0.918 ms
64 bytes from 192.168.111.65: icmp_seq=12 ttl=64 time=1.16 ms
64 bytes from 192.168.111.65: icmp_seq=13 ttl=64 time=0.942 ms
64 bytes from 192.168.111.65: icmp_seq=14 ttl=64 time=0.924 ms
64 bytes from 192.168.111.65: icmp_seq=15 ttl=64 time=0.787 ms 🦷
64 bytes from 192.168.111.65: icmp_seq=16 ttl=64 time=1.29 ms

Tous les tests sont fonctionnels donc nous avons réussi les interconnexions entre les machines client et la machine serveur en testant les différents VLANS !

## Diagramme des tâches

https://docs.google.com/spreadsheets/d/1ebEXhsJLcYOcEPvgEmgHis\_OGqmUWFYM7L8SXV dGKE0/edit#gid=1115838130