Mission 5 : Serveur Base de données MariaDB

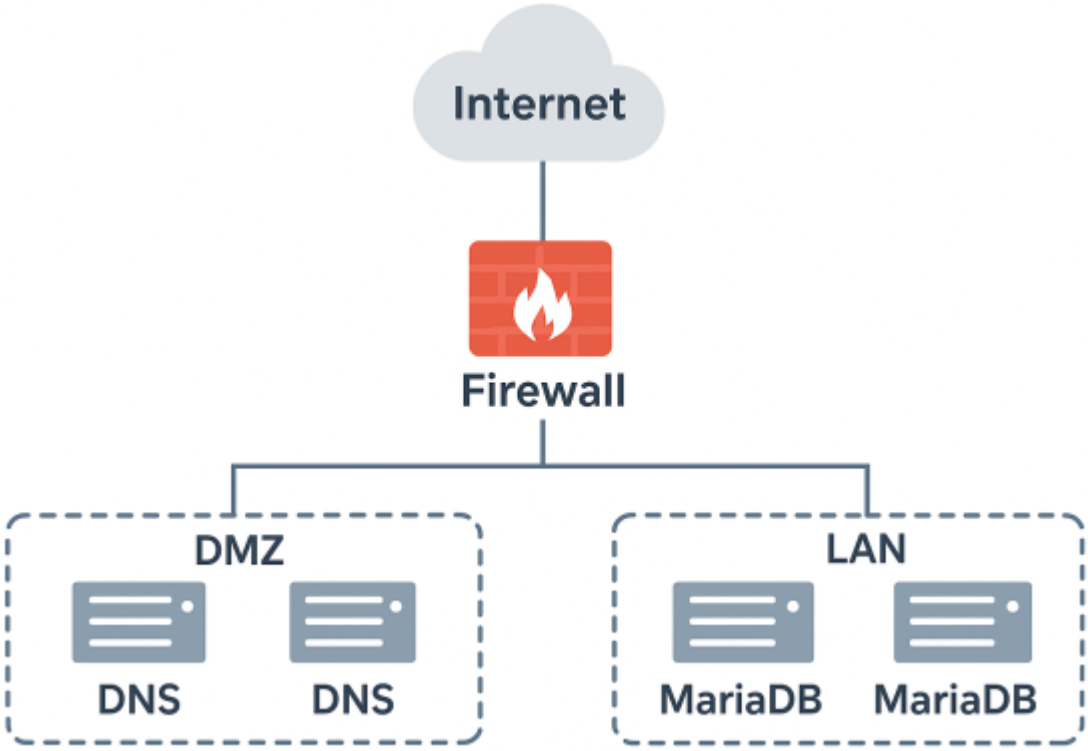
Pré-requis

Afin de pouvoir réaliser cette mission il faut :

- Un serveur sous Debian 13
- Un routeur sous Debian 13
- La configuration IP des deux machines conforme au cahier des charges,
- La configuration des deux Base de données conforme au cahier des charges,

Introduction

Nous allons installer deux serveur de base de données sur mariaDB en mysql sur notre reseau LAN et non pas dans la DMZ car la DMZ est accessible a tout le monde. Ce qui veut dire que tout le monde aurait accés a notre base de données et ce n'est pas ce que l'on veut



Avantage de les mettres dans le LAN:

* Si les serveurs de base de données ne sont pas dans la DMZ c'est pour rajouter une couche de sécuriter pour qu'ils ne soit pas accessible par tout le monde.

> * Le fait d'avoir la base de données ainsi que le serveur web dans le LAN nous permet de faire une replication inverse de la DMZ vers le LAN

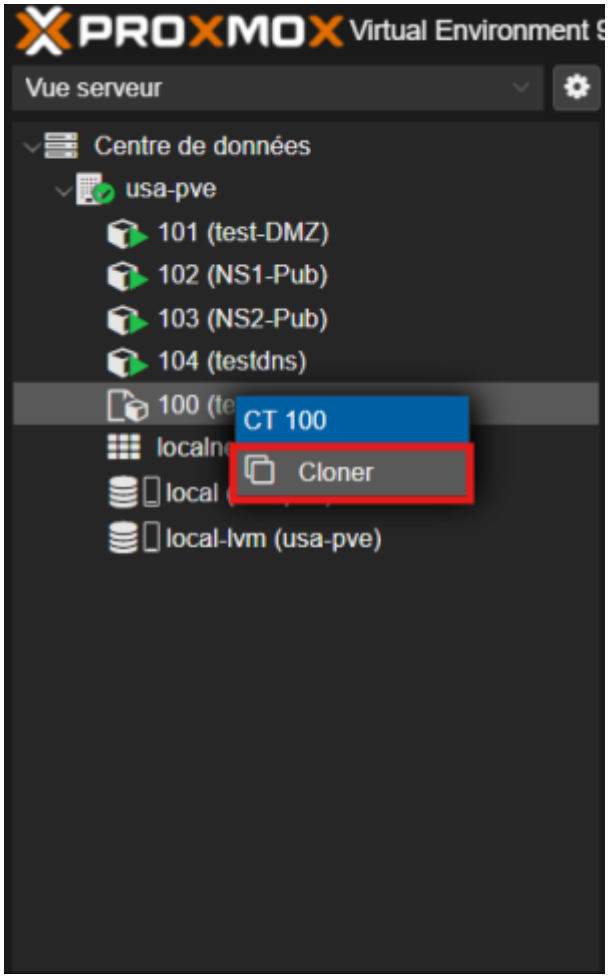
Cahier des charges

- Les derniers octets des IP des deux machines seront **.33** et **.34**.
- Les noms d'hôte des machines seront **priv-db1** et **priv-db2**.
- Création d'un compte administrateur « **admin** » (mot de passe **drowssap**) ayant tous les droits sur toutes les bases de données avec la permission de modifier les droits des autres utilisateurs.
- Création d'une base de données **gsb**.
- Création d'un compte **gsb** (mot de passe **drowssap**) ayant uniquement les droits sur la base de données **gsb**.
- Créer un compte « test » dont le schéma d'authentification utilisé est ed25519 grâce au plugin d'authentification **ed25519**.
- Les comptes **gsb** et **admin** doivent pouvoir se connecter au serveur de bases de données depuis une machine distante.

Configuration premier conteneur

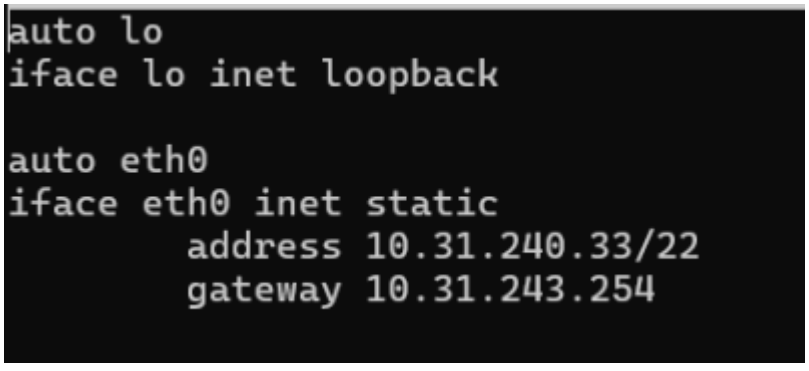
Création et parametrage reseau du conteneur BDD-1

Pour commencer il nous faut copier notre conteneur template



Après avoir cloné notre Template il fallait modifier les paramètres IP pour qu'il soit conforme au cahier des charges. Le cahier des charges nous demande de mettre nos Base de données dans le reseau LAN qui dans notre cas et le 1er et 2eme sous reseau et donc en 10.31.240.33/34 Je suis donc aller dans le répertoire etc/network pour remplir le fichier interfaces et ainsi mettre ma nouvelle adresse IP :

```
> nano /etc/network/interfaces
```

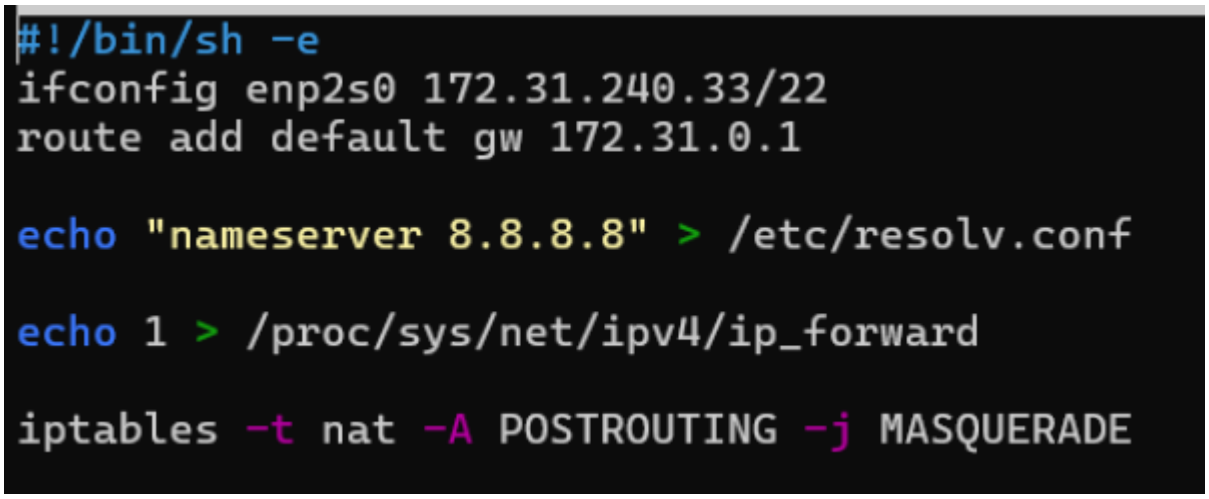



```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
        address 10.31.240.33/22
```

```
        gateway 10.31.243.254
```

Pour finir la configuration réseau je suis retourner dans le répertoire etc mais cette fois-ci dans le dossier rc.local pour y implanter tout les paramètre réseau nécessaire au bon fonctionnement du centeneure




```
#!/bin/sh -e
ifconfig eth0 10.31.240.33/22 up
route add default gw 172.31.0.1

echo "nameserver 8.8.8.8" > /etc/resolv.conf

echo 1 > /proc/sys/net/ipv4/ip_forward

iptables -t nat -A POSTROUTING -j MASQUERADE
```

Important:

Faire un systemctl restart networking a fin de redémarrer le service pour que la machine prennent les nouvelle configuration réseau

Installation et parametrage de MariaDB

Installation de MariaDB

```
apt update
apt install mariadb-server php-mysql
```

une fois installé pour rentrer dans MariaDB il faut se connecter avec le compte root

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

Nous allons créer 2 utilisateurs conforme au cahier des charges. Un utilisateur Admin qui lui a les droits sur toute les base de données, et un utilisateur gsb qui a les droits que sur une base de données nommée gsb sur la quelle il a été attribuée

voir cahier des charges

Création de la base de donnée

Commencont par créer la base de donnée souligner dans le cahier des charges. Pour ce faire taper la commande suivante :

```
MariaDB [(none)]> ceate database gsb
```

```
MariaDB [(none)]> create database gsb;
Query OK, 1 row affected (0.000 sec)
```

Cette ligne va permettre de créer la Base de donnée nommé gsb

Création des Utilisateurs

Une fois dans MariaDB il nous suffit de taper les commandes :

```
MariaDB [(none)]>  create user 'admin'@'%' identified by 'drowssap';
MariaDB [(none)]>  create user 'gsb'@'%' identified via 'ed25519';
```

```
MariaDB [(none)]> create user 'admin'@'%' identified by 'drowssap';
Query OK, 0 rows affected (0.021 sec)
```

```
MariaDB [(none)]> create user 'gsb'@'%' identified via 'ed25519';
Query OK, 0 rows affected (0.006 sec)
```

Ces commandes vont créer 2 utilisateurs. L'utilisateur admin et gsb.

On peut voir que les commande ne sont pas les mêmes

Create user : Sert a créer un utilisateur
'admin'@'%' : veut dire que l'on créer l'utilisateur nommée Admin avec comme option “%” ce qui signifie que le compte admin peut se conneceter a la base de donnée a distance autrement l'option serait “Localhost”
'gsb'@'%' : veut dire que l'on créer l'utilisateur nommée Admin avec comme option “%” ce qui signifie que le compte admin peut se conneceter a la base de donnée a distance autrement l'option serait “Localhost”
Identified by 'drowssap' : Signifie que le mot de passe de l'utilisateur est drowssap
Identified via 'ed25519' : Signifie que l'utilisateur est obligé de passer par le plugin ed25519 ce qui va authentifier l'uitilisateur via une clé publique plutot qu'un mot de passe

Il est important de faire passer l'utilisateur gsb par le plugin car ce plugin permet d'authentifier l'utilisateur sans que le mot de passe de celui-ci soit stocké dans la base de donnés donc si la base de donnés ce fait craqué il ne pourront pas se connecter a la base de donnée sur la quelle l'utilisateur gsb est tout permis

Parametrage des Utilisateurs

Aprés avoir créé nos utilisateurs il va falloir les parametrer. Pour ce faire nous allons taper ces commandes :

```
MariaDB [(none)]>  grant all privileges on *.* to 'admin'@'%' with grant
option;
MariaDB [(none)]>  grant all privileges on gsb.* to 'gsb'@'%' with grant
option;
```

```
MariaDB [(none)]> grant all privileges on *.* to 'admin'@'%' with grant option;
Query OK, 0 rows affected (0.015 sec)
```

```
MariaDB [(none)]> grant all privileges on gsb.* to 'gsb'@'%' with grant option;
Query OK, 0 rows affected (0.020 sec)
```

GRANT : est l'ordre pour accorder des droits aux utilisateurs

All privileges : accorde tout les droit possible (SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, etc.)
on *.* : veut dire que tout les droit possible seront sur toutes les bases de données existante
on gsb.* : veut dire que tout les droit possible seront que sur la base de données gsb
With grant option: Permet à cet utilisateur de donner à d'autres utilisateurs les mêmes droits qu'il a reçus

Ca veut dire que l'utilisateur Admin a tout les droit sur toute les base de données contrairement a l'utilisateur gsb qui lui a tout les droits uniquement sur la base de données nommée gsb ce qui est trés important en terme de cybersecurité puisque si le serveur de base de donnée se fait perser sur le compte Admin, les pirates n'auront pas accées a la base de données gsb

Fichier de configuration

Aprés avoir créé et parametrer tout nos utilisateur, le serveur n'est quand même pas accessible car comme on peut le voir grace a la commande :

```
netstat -nat
```

```
root@BDD-1:~# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 ::1:25                  :::*                    LISTEN
tcp6       0     36 10.31.240.33:22         10.31.243.254:40798     ESTABLISHED
```

Le port 3306 qui est le port de MariaDB n'autorise que l'adresse de Loopback ce qui fait que si l'on veut interroger la base de donnée depuis l'exterieur la base de donnée ne recevra jamais les requettes.
Pour changer ca il faut aller dans le fichie de configuration 50-serveur.cnf qui se trouve dans :

```
nano /etc/mysql/mariadb.conf.d/50-serveur.cnf
```

Dans lequelle nous devant changer l'adresse de loopback par 0.0.0.0 pour que la base de données accepte les requettes de tout le monde

```
#
# These groups are read by MariaDB server.
# Use it for options that only the server (but not clients) should see

# this is read by the standalone daemon and embedded servers
[server]

# this is only for the mariadbd daemon
[mariadbd]

#
# * Basic Settings
#

#user                   = mysql
pid-file                = /run/mysqld/mysqld.pid
basedir                 = /usr
#datadir                = /var/lib/mysql
#tmpdir                 = /tmp

# Broken reverse DNS slows down connections considerably and name resolve is
# safe to skip if there are no "host by domain name" access grants
#skip-name-resolve

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0
```

Une fois cela fait ne surout pas oublier de faire un systemctl restart mariadb pour que toutes les modification soit pris en compte

le systemctl fait, refaire la commande netstat -nat ce qui nous fera apparaitre l'adresse 0.0.0.0 sur le port 3306 ce qui voudra dire que le serveur de base de donnée recevra toutes les requettes qu'on lui envoie

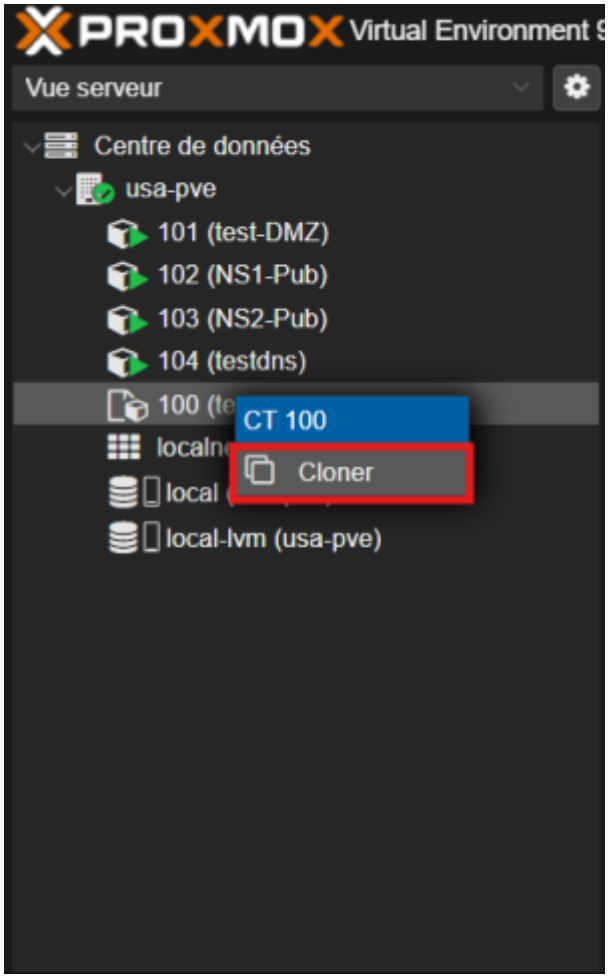
```
root@BDD-1:~# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 ::1:25                  :::*                    LISTEN
tcp6       0     52 10.31.240.33:22         10.31.243.254:54683     ESTABLISHED
root@BDD-1:~# mysql -u root -p
```

Si le resultat est le suivant alors le serveur est terminé d'être configurer et est utilisable

Configuration deuxieme conteneur

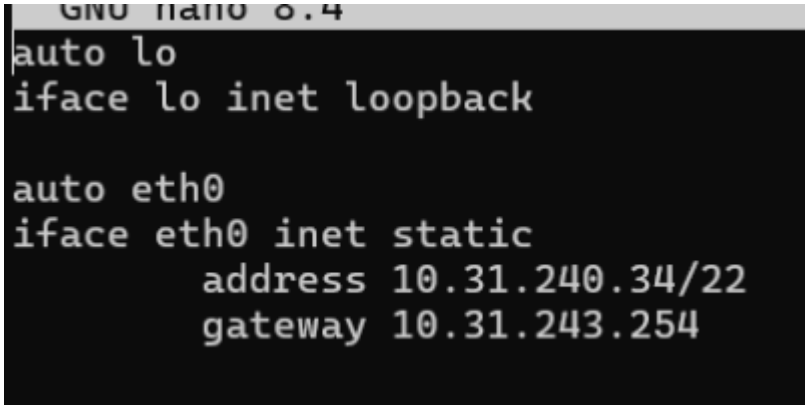
Création et parametrage reseau du conteneur BDD-2

Pour commencer il nous faut copier notre conteneur template



Après avoir cloné notre Template il fallait modifier les paramètres IP pour qu'il soit conforme au cahier des charges. Le cahier des charges nous demande de mettre nos Base de données dans le reseau LAN qui dans notre cas et le 1er et 2eme sous reseau et donc en 10.31.240.33/34 Je suis donc aller dans le répertoire etc/network pour remplir le fichier interfaces et ainsi mettre ma nouvelle adresse IP :

```
> nano /etc/network/interfaces
```

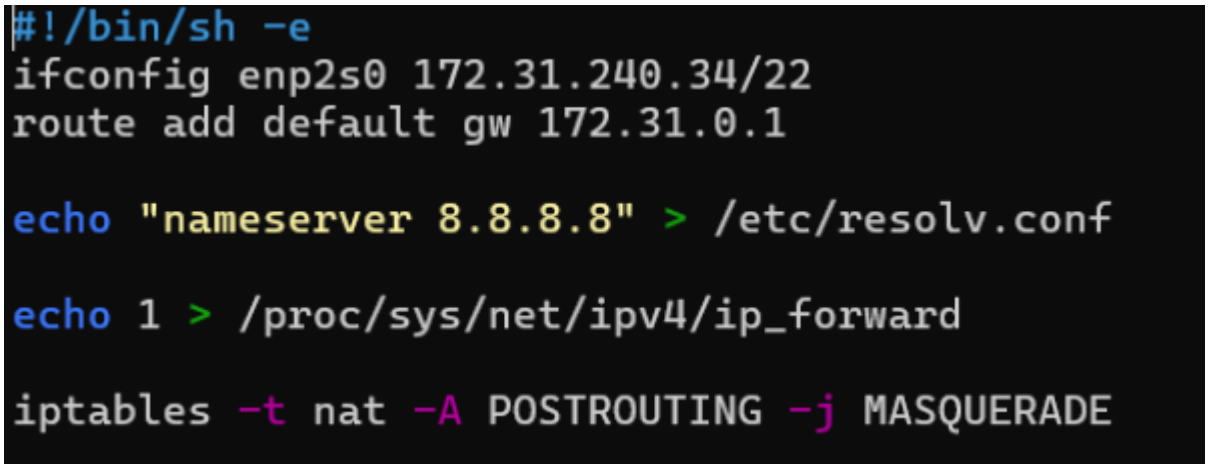



```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
        address 10.31.240.34/22
```

```
        gateway 10.31.243.254
```

Pour finir la configuration réseau je suis retourner dans le répertoire etc mais cette fois-ci dans le dossier rc.local pour y implanter tout les paramètre réseau nécessaire au bon fonctionnement du centeneure




```
#!/bin/sh -e
ifconfig eth0 10.31.240.34/22 up
route add default gw 172.31.0.1

echo "nameserver 8.8.8.8" > /etc/resolv.conf

echo 1 > /proc/sys/net/ipv4/ip_forward

iptables -t nat -A POSTROUTING -j MASQUERADE
```

Important:

Faire un systemctl restart networking a fin de redémarrer le service pour que la machine prennent les nouvelle configuration réseau

Installation et parametrage de MariaDB

Installation de MariaDB

```
apt update
```

```
apt install mariadb-server php-mysql
```

une fois installé pour rentrer dans MariaDB il faut se connecter avec le compte root

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

Nous allons créer 2 utilisateurs conforme au cahier des charges. Un utilisateur Admin qui lui a les droits sur toute les base de données, et un utilisateur gsb qui a les droits que sur une base de données nommée gsb sur la quelle il a été attribuée

voir cahier des charges

Création de la base de donnée

Commencont par créer la base de donnée souligner dans le cahier des charges. Pour ce faire taper la commande suivante :

```
MariaDB [(none)]> ceate database gsb
```

```
MariaDB [(none)]> create database gsb;
Query OK, 1 row affected (0.000 sec)
```

Cette ligne va permettre de créer la Base de donnée nommé gsb

Création des Utilisateurs

Une fois dans MariaDB il nous suffit de taper les commandes :

```
MariaDB [(none)]>  create user 'admin'@'%' identified by 'drowssap';
MariaDB [(none)]>  create user 'gsb'@'%' identified via 'ed25519';
```

```
MariaDB [(none)]> create user 'admin'@'%' identified by 'drowssap';
Query OK, 0 rows affected (0.021 sec)
```

```
MariaDB [(none)]> create user 'gsb'@'%' identified via 'ed25519';
Query OK, 0 rows affected (0.006 sec)
```

Ces commandes vont créer 2 utilisateurs. L'utilisateur admin et gsb.

On peut voir que les commande ne sont pas les mêmes

Create user : Sert a créer un utilisateur
'admin'@'%' : veut dire que l'on créer l'utilisateur nommée Admin avec comme option “%” ce qui signifie que le compte admin peut se conneceter a la base de donnée a distance autrement l'option serait “Localhost”
'gsb'@'%' : veut dire que l'on créer l'utilisateur nommée Admin avec comme option “%” ce qui signifie que le compte admin peut se conneceter a la base de donnée a distance autrement l'option serait “Localhost”
Identified by 'drowssap' : Signifie que le mot de passe de l'utilisateur est drowssap
Identified via 'ed25519' : Signifie que l'utilisateur est obligé de passer par le plugin ed25519 ce qui va authentifier l'uitilisateur via une clé publique plutot qu'un mot de passe

Il est important de faire passer l'utilisateur gsb par le plugin car ce plugin permet d'authentifier l'utilisateur sans que le mot de passe de celui-ci soit stocké dans la base de donnés donc si la base de donnés ce fait craqué il ne pourront pas se connecter a la base de donnée sur la quelle l'utilisateur gsb est tout permis

Parametrage des Utilisateurs

Aprés avoir créé nos utilisateurs il va falloir les parametrer. Pour ce faire nous allons taper ces commandes :

```
MariaDB [(none)]>  grant all privileges on *.* to 'admin'@'%' with grant
option;
MariaDB [(none)]>  grant all privileges on gsb.* to 'gsb'@'%' with grant
option;
```

```
MariaDB [(none)]> grant all privileges on *.* to 'admin'@'%' with grant option;
Query OK, 0 rows affected (0.015 sec)
```

```
MariaDB [(none)]> grant all privileges on gsb.* to 'gsb'@'%' with grant option;
Query OK, 0 rows affected (0.020 sec)
```

GRANT : est l'ordre pour accorder des droits aux utilisateurs
All privileges : accorde tout les droit possible (SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, etc.)
on *.* : veut dire que tout les droit possible seront sur toutes les bases de données existante
on gsb.* : veut dire que tout les droit possible seront que sur la base de données gsb

With grant option: Permet à cet utilisateur de donner à d'autres utilisateurs les mêmes droits qu'il a reçus

Ca veut dire que l'utilisateur Admin a tout les droit sur toute les base de données contrairement a l'utilisateur gsb qui lui a tout les droits uniquement sur la base de données nommée gsb ce qui est trés important en terme de cybersecurité puisque si le serveur de base de donnée se fait perser sur le compte Admin, les pirates n'auront pas accées a la base de données gsb

Fichier de configuration

Aprés avoir créé et parametrer tout nos utilisateur, le serveur n'est quand même pas accessible car comme on peut le voir grace a la commande :

```
netstat -nat
```

```
root@BDD-1:~# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 ::1:25                  :::*                    LISTEN
tcp6       0     36 10.31.240.33:22         10.31.243.254:40798     ESTABLISHED
```

Le port 3306 qui est le port de MariaDB n'autorise que l'adresse de Loopback ce qui fait que si l'on veut interroger la base de donnée depuis l'exterieur la base de donnée ne recevra jamais les requettes.
Pour changer ca il faut aller dans le fichie de configuration 50-serveur.cnf qui se trouve dans :

```
nano /etc/mysql/mariadb.conf.d/50-serveur.cnf
```

Dans lequelle nous devant changer l'adresse de loopback par 0.0.0.0 pour que la base de données accepte les requettes de tout le monde

```
#
# These groups are read by MariaDB server.
# Use it for options that only the server (but not clients) should see

# this is read by the standalone daemon and embedded servers
[server]

# this is only for the mariadbd daemon
[mariadbd]

#
# * Basic Settings
#

#user                   = mysql
pid-file                = /run/mysqld/mysqld.pid
basedir                 = /usr
#datadir                = /var/lib/mysql
#tmpdir                 = /tmp

# Broken reverse DNS slows down connections considerably and name resolve is
# safe to skip if there are no "host by domain name" access grants
#skip-name-resolve

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0
```

Une fois cela fait ne surout pas oublier de faire un systemctl restart mariadb pour que toutes les modification soit pris en compte

le systemctl fait, refaire la commande netstat -nat ce qui nous fera apparaitre l'adresse 0.0.0.0 sur le port 3306 ce qui voudra dire que le serveur de base de donnée recevra toutes les requettes qu'on lui envoie

```
root@BDD-2:~# netstat -nat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:3306            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 ::1:25                  :::*                    LISTEN
tcp6       0     60 10.31.240.2:22          10.31.243.254:55961     ESTABLISHED
tcp6       0     36 10.31.240.34:22         10.31.243.254:57161     ESTABLISHED
root@BDD-2:~# cd mariadb.conf.d/client_loop: send disconnect: Connection reset
PS C:\Users\gauti>
```

Si le resultat est le suivant alors le serveur est terminé d'être configurer et est utilisable