

Compétences	OO2 - Attribuer des rôles/autorisations pour garantir la protection et la sécurité des données.		
Objectifs	1	Création de compte	1
	2	Utilisation du compte	2
	3	Suppression de compte	2
	4	Attribution des droits	2
	5	Moyens de connaître qui a quel droit	3
	6	Suppression des droits	3
	7	Pour tester ...	4
Durée estimée	15 min		
Répertoire de travail	.\oo2-dcl		
Fichiers sources			
A produire			
Référence	http://dev.mysql.com/doc/refman/5.7/en/account-management-sql.html		

1 Création de compte

1.1 Création d'un compte au niveau du serveur local uniquement :

```
CREATE USER 'jeffrey'@'localhost' IDENTIFIED BY 'new_password';
```

Après cette instruction SQL, le compte d'un utilisateur est créé, donc connu du serveur. Toutefois, cet utilisateur n'aura pour l'instant aucun autre droit que de se connecter sur le serveur.

1.2 Création d'un compte au niveau d'un domaine spécifié :

```
CREATE USER 'jeffrey'@'%.etnlnet.local' IDENTIFIED BY 'new_password';
```

1.3 Création d'un compte au niveau de tous les domaines :

```
CREATE USER 'jeffrey'@'%' IDENTIFIED BY 'new_password';
```

2 Utilisation du compte

```
"%easyrootpath%\binaries\mysql\bin\mysql.exe" --user=test01 --password <db>
```

3 Suppression de compte

3.1 Suppression d'un compte sur le serveur local :

```
DROP USER 'jeffrey'@'localhost';
```

3.2 Suppression d'un compte au niveau d'un domaine spécifique :

```
DROP USER 'jeffrey'@'%etmlnet.local';
```

3.3 Suppression d'un compte au niveau de tous les domaines :

```
DROP USER 'jeffrey'@'%';
```

4 Attribution des droits

4.1 Syntaxe pour l'attribution de droits à un compte :

```
GRANT <priv1,priv2,...> ON <dbname>.<dbobject> TO <user>'@'<hostname>' [WITH GRANT OPTION];
```

4.2 Attribution de tous les droits à un compte au niveau du serveur :

```
GRANT ALL ON *.* TO 'someuser'@'localhost' WITH GRANT OPTION;
```

4.3 Attribution de droits de lecture et écriture, à un compte, au niveau du serveur :

```
GRANT SELECT, INSERT, UPDATE, DELETE ON *.* TO 'someuser'@'localhost';
```

4.4 Attribution de droits d'accès seulement, à un compte, au niveau du serveur :

```
GRANT USAGE ON *.* TO 'someuser'@'localhost';
```

4.5 Attribution de droits de lecture, à un compte, au niveau du serveur :

```
GRANT SELECT ON *.* TO 'someuser'@'localhost';
```

4.6 Suppression de tous les droits à un compte au niveau du serveur :

```
REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'someuser'@'localhost';
```

4.7 Suppression de tous les droits à un compte au niveau d'une BD :

```
REVOKE ALL PRIVILEGES ON dbname.* FROM 'someuser'@'localhost';  
REVOKE GRANT OPTION ON dbname.* FROM 'someuser'@'localhost';
```

5 Moyens de connaître qui a quel droit

5.1 Au niveau du serveur, qui a le droit de lire (SELECT) ?

```
SELECT USER, HOST, PASSWORD, SELECT_PRIV FROM MYSQL.USER;
```

5.2 Au niveau du serveur, qui a le droit de lire dans quelle BD ?

```
SELECT USER, DB, SELECT_PRIV FROM MYSQL.DB ;
```

5.3 Au niveau du serveur, quels sont les droits d'un utilisateur en local ?

```
SHOW GRANTS FOR lecteur@localhost;
```

5.4 Au niveau de l'utilisateur courant, quels sont ses droits en général ?

```
SHOW GRANTS;
```

```
SHOW GRANTS FOR CURRENT_USER;
```

6 Suppression des droits

6.1 Syntaxe pour la suppression des droits à un compte :

```
REVOKE priv_type [(column_list)] [, priv_type [(column_list)]] ON [object_type] priv_level  
FROM user [, user] ...
```

Ou ...

```
REVOKE ALL PRIVILEGES, GRANT OPTION FROM user [, user] ...
```

6.2 Exemple pour la suppression des droits à un compte :

```
REVOKE INSERT ON mydb.* FROM 'jeffrey'@'localhost';
```

7 Pour tester ...

7.1 Sur le serveur :

```
CREATE USER 'user1'@'%etmlnet.local' IDENTIFIED BY 'user1';  
GRANT ALL ON *.* TO 'user1'@'%etmlnet.local' WITH GRANT OPTION;
```

7.2 Modification de my.ini du côté du serveur :

```
# ---- mettre en commentaire la ligne suivante avec le caractère # à gauche  
#bind-address      = 127.0.0.1  
# ---- redémarrer le serveur
```

7.3 Connexion depuis un client :

```
"%easyrootpath%\binaries\mysql\bin\mysql.exe"  
--host=<nom machine etml>.etmlnet.local  
--user=user1 --password=user1  
<db>
```

7.4 Exemple

REM

REM Sur le serveur remote, le compte "essaieleve" a été créé de la manière suivante

REM

```
CREATE USER 'essaieleve'@'%' IDENTIFIED WITH mysql_native_password AS  
'***';GRANT SELECT, INSERT, UPDATE, DELETE, FILE ON *.* TO 'essaieleve'@'%'  
REQUIRE NONE WITH MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0  
MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0;GRANT ALL PRIVILEGES ON `dbtest`.*  
TO 'essaieleve'@'%';
```

REM Sur la machine locale, lorsque l'on veut se connecter sur le serveur remote ...

```
"%easyrootpath%\eds-binaries\dbserver\mysql5711x86x160823145809\bin\mysql.exe"  
--user=essaieleve  
--password=.etml-  
--host=inf-N103-M502  
dbtest
```