

SGBD 1

SupNum

Gestion des utilisateurs

Gestion des comptes utilisateurs

- La gestion des utilisateurs de **MySQL** se fait entièrement grâce à des tables dédiées qui appartiennent au schéma système **mysql**
- Grâce à des commandes SQL permettant de créer, modifier et supprimer les comptes utilisateurs et leurs droits.
- La table **user** contient la liste des comptes utilisateurs.
 - Les colonnes **host** (le nom d'hôte) et **user** (le nom) (colonnes représentent la clé primaire de la table.)
password (le mot de passe du compte)
 - Les droits globaux de l'utilisateur, comme le droit de faire des insertions (INSERT), des lectures (SELECT)...

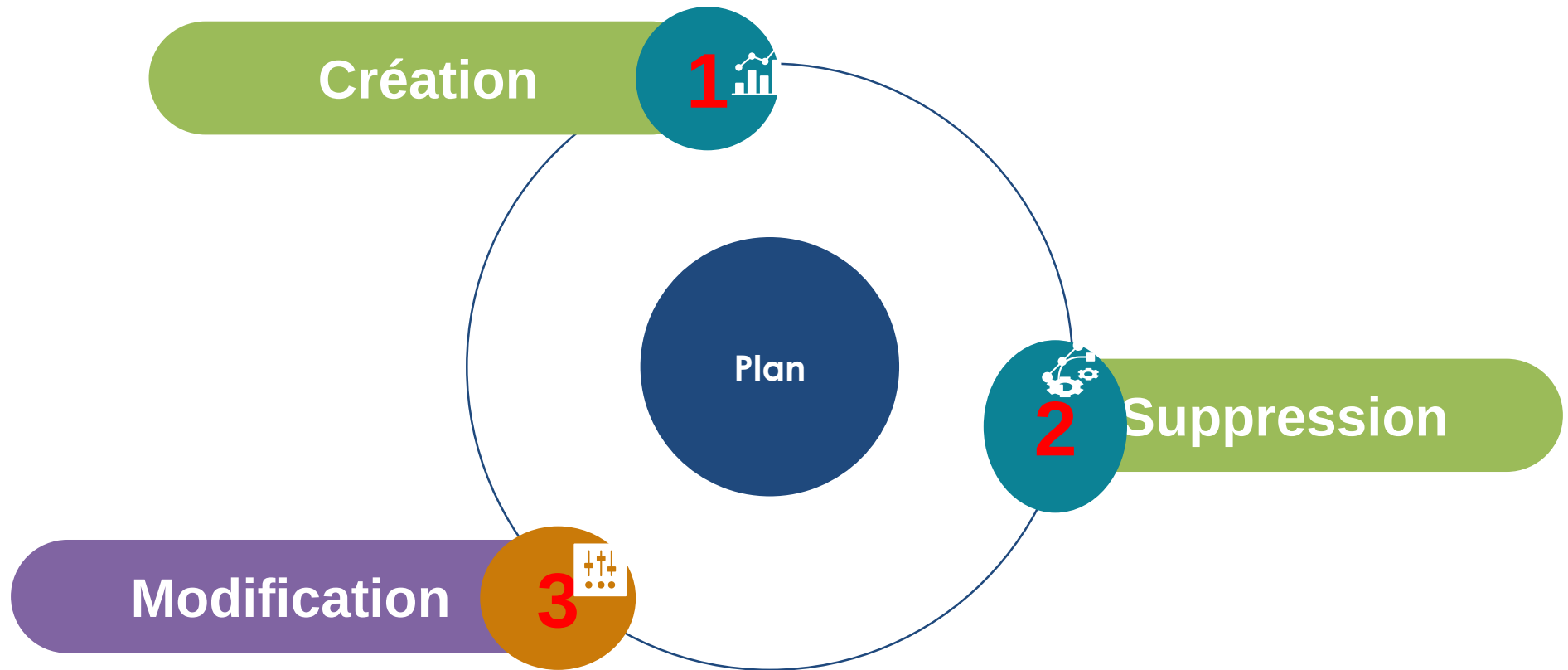
Gestion des comptes utilisateurs

- La table **db** contient les droits spécifiques à une base de données (ou schémas) de chaque utilisateur.
- La table **tables_priv** contient les droits spécifiques à une table ou à une vue.
- La table **columns_priv** contient les droits spécifiques à une colonne.
- La table **procs_priv** contient les droits pour les routines stockées, c'est-à-dire les procédures et les fonctions stockées.

1. **Information_schema** : stocke les informations sur toutes les bases de données. Les tables, les colonnes, le type des colonnes, etc.
2. **performance_schema** : stocke des informations sur les actions effectuées sur le serveur (temps d'exécution, temps d'attente dus aux verrous, etc.)
3. **test** : base de test automatiquement créée (vide au début)
4. **MySQL** : contient de nombreuses informations sur le serveur. Elle stocke **les utilisateurs (user) et leurs privilèges**. (db, *tables_priv*, *columns_priv*, *proc_priv*),

Allez S.V.P à la console
MySQL !

Création et modification des utilisateurs



Création et suppression

-- Création

```
CREATE USER 'login'@'hote' [IDENTIFIED BY 'mot_de_passe'];
```

-- Suppression

```
DROP USER 'login'@'hote';
```

Exemple de création

```
CREATE USER 'etudiant'@'localhost' IDENTIFIED BY 'cheikh';
```

Exemple de suppression

```
DROP USER 'etudiant'@'localhost';
```

Création et suppression

Utilisateur

L'utilisateur est donc défini par **deux éléments** :

- son login ;
- l'hôte à partir duquel il se connecte.

Login

Le login est un simple identifiant. Vous pouvez le choisir comme vous voulez. Il n'est pas obligatoire de l'entourer de guillemets, sauf s'il contient des caractères spéciaux comme — ou @

Hôte

L'hôte est **l'adresse à partir de laquelle l'utilisateur va se connecter**. Si l'utilisateur se connecte à partir de la machine sur laquelle le serveur MySQL se trouve, on peut utiliser 'localhost'. Sinon, on utilise en général une adresse IP ou un nom de domaine.

Exemples :

```
CREATE USER 'ahmed'@'localhost' IDENTIFIED BY 'ahmedmed';  
CREATE USER 'fatma'@'194.28.12.4' IDENTIFIED BY 'fatou1';  
CREATE USER 'sidi'@'iscae.mr' IDENTIFIED BY 'iscae';
```

Connexion à plusieurs hôtes différents (le joker %)

- Possibilité de se connecter à partir de plusieurs hôtes différents
- Sans devoir créer un utilisateur par hôte
- On peut préciser des noms d'hôtes partiels
- Ou permettre à l'utilisateur de se connecter à partir de n'importe quel hôte.

Exemples :

-- mohamed peut se connecter à partir de n'importe quel hôte dont l'adresse IP commence par 194.28.12.

```
CREATE USER 'mohamed'@'194.28.12.%' IDENTIFIED BY 'SupNum2024';
```

-- mariem peut se connecter à partir de n'importe quel hôte du domaine iscae.mr

```
CREATE USER 'mariem'@'%iscae.mr' IDENTIFIED BY 'adminsупnum';
```

-- amadou peut se connecter à partir de n'importe quel hôte

```
CREATE USER 'amadou'@'%' IDENTIFIED BY 'amda$123';
```

Remarque :

Notez que, si vous ne précisez pas d'hôte, c'est comme si vous autorisiez tous les hôtes
'cheikh'@'%' est équivalent à 'cheikh'

Renommer un utilisateur

Pour modifier l'identifiant d'un utilisateur (login et/ou hôte), on peut utiliser **RENAME USER** ancien_utilisateur **TO** nouvel_utilisateur

Exemple :

RENAME USER cheikh **TO** dhib

Mot de passe

- Le mot de passe de l'utilisateur est donné par la clause **IDENTIFIED BY**
- N'est pas obligatoire, auquel cas l'utilisateur peut se connecter sans donner de mot de passe.
- Du point de vue de la sécurité, il est conseiller d'**Évitez au maximum les utilisateurs sans mot de passe.**
- N'est pas stocké tel quel dans la table *mysql.user*. Il est d'abord crypté (hashé), et c'est cette valeur cryptée qui est stockée.

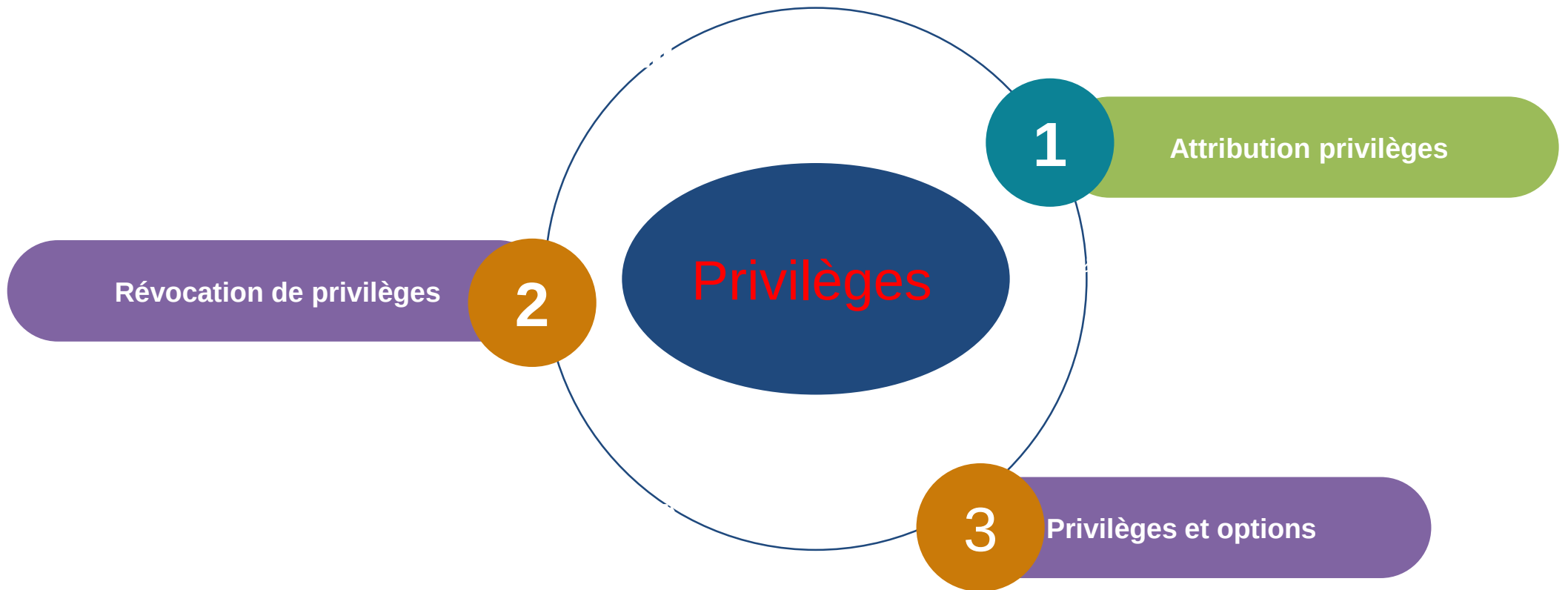
Modifier le mot de passe

Pour modifier le mot de passe d'un utilisateur, on peut utiliser la commande ***SET PASSWORD. Il faut soit***

- avoir les privilèges nécessaires, ou
- Etre connecté avec l'utilisateur dont on veut changer le mot de passe.

Exemple : SET PASSWORD FOR 'etudiant'@'localhost' = 'DSI2024';

Gestion des utilisateurs – Privilèges



Privilèges des utilisateurs - Introduction

- On se connecte sur MySQL avec un utilisateur (exemple : **root**)
- Chaque utilisateur possède un ensemble de privilèges relatifs aux données stockées sur le serveur
- Les privilèges peuvent être la **sélection** des données, la **modification**, la **création** des objets, etc.
- Les privilèges peuvent être attribué par niveau : **base de données**, **tables**, **colonnes**, **procédures**, etc.

Privilèges des utilisateurs : stockage

- *User* : on y stocke les privilèges globales
- *db* : privilèges au niveau des bases de données.
- *tables_priv* : privilèges au niveau des tables.
- *columns_priv* : privilèges au niveau des colonnes.
- *proc_priv* : privilèges au niveau des routines (procédures et fonctions stockées).

Privilèges des utilisateurs : types de droits

- Lors de la création, pas de droits (sauf la connexion)
- Donc pas de possibilité de faire des actions
- **CRUD : Insert, select, update et delete**
- **Privilèges sur les bases de données, les tables et les vues : create table, create temporary table, create view ((il faut également avoir le privilège SELECT sur les colonnes sélectionnées par la vue), ALTER (modification d'une table avec alter table), DROP (suppression des tables, vues et bases de données)**
- **Utilisateurs : CREATE USER, drop user, rename user, set password)**
- **Autres : gestion des procédures, déclencheurs, etc.**

Privilèges des utilisateurs – Niveaux

Commande	Niveau d'application	Table de stockage
* * .	Toutes les bases de données, tous les objets	Mysql.user
*	1. Sans sélectionner une bases : équivalent à *.* 2. Avec la sélection d'une base : sur tous les objets de cette base	1. MySQL.user 2. Mysql.db
Nom_base.*	Privilège de base de données : s'applique à tous les objets de la base nom_bdd	mysql.db
nom_bdd.nom_table	Privilège de table	mysql.tables_priv
nom_table (Privilège de table)	s'applique à la table nom_table de la base de données dans laquelle on se trouve, sélectionnée au préalable avec USE nom_bdd	mysql.tables_priv
Privilège sur colonnes	Sur des colonnes d'une table	mysql.columns_priv
Nom_bdd.routine	Pour les procédures, fonctions ,etc.	Mysql.procs_priv

Privilèges des utilisateurs : attribution et suppression des privilèges

1. Attribution : commande GRANT

- Pour attribuer une privilège, l'utilisateur doit avoir le privilège **Grant option**

- **Syntaxe :**

GRANT privilege [(liste_colonnes)] [, privilege
[(liste_colonnes)], ...] **ON** [type_objet] niveau_privilege
TO utilisateur [IDENTIFIED BY mot_de_passe];

- ☐ privilege : le privilège à accorder à l'utilisateur (SELECT, CREATE VIEW, EXECUTE...);
- ☐ (liste_colonnes) : facultatif - liste des colonnes auxquelles le privilège s'applique ;
- ☐ niveau_privilege : niveau auquel le privilège s'applique (*.*, nom_bdd.nom_table...);
- ☐ type_objet : en cas de noms ambigus, il est possible de préciser à quoi se rapporte le niveau, TABLE ou PROCEDURE

```
CREATE USER 'cheikh'@'moi' IDENTIFIED BY  
'MeToo';  
GRANT SELECT,  
      UPDATE (PLNOM, PLPRENOM),  
      DELETE,  
      INSERT  
ON vols.pilote  
TO 'moi'@'localhost';
```

```
GRANT SELECT ON TABLE vols.avion TO  
'moi'@'localhost';
```

Privilèges des utilisateurs-Suppression de privilèges

Révocation: commande **REVOKE**

- ☐ On utilise la commande **REVOKE**
- ☐ On peut retirer plusieurs privilèges à la fois
- ☐ Syntaxe :

```
REVOKE privilege [, privilege, ...]  
ON niveau_privilege  
FROM utilisateur;
```

Exemple :

```
REVOKE DELETE  
ON vols.pilote  
FROM 'moi'@'localhost';
```

Privilèges des utilisateurs : Les privilèges ALL, USAGE et GRANT OPTION

ALL (ALL PRIVILEGES): sert à accorder tous les droits à l'utilisateur (Il faut évidemment préciser le niveau auquel tous les droits sont accordés)

Usage : n'attribue aucun privilège, peut être utile pour modifier l'utilisateur avec GRANT

GRANT OPTION : sert à accorder tous les droits à l'utilisateur (Il faut évidemment préciser le niveau auquel tous les droits sont accordés)

```
GRANT ALL ON vols.vol TO 'moi'@'localhost';
```

```
GRANT SELECT, UPDATE, INSERT, DELETE,  
GRANT OPTION ON vols.* TO 'moi'@'localhost'  
IDENTIFIED BY 'cheikh';
```

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

```
GRANT SELECT, UPDATE, INSERT, DELETE  
ON vols.* TO 'moi'@'localhost' IDENTIFIED BY  
'cheikh'  
WITH GRANT OPTION;
```