esprit
Ecole Supérieure Privée
d'Ingénierie et de Technologies

Chapitre 3 : Administrer la sécurité Utilisateur

Objectifs du cours

- Créer des utilisateurs des rôles et des profils
- Affecter des privilèges à des rôles
- Distinguer les privilèges systèmes des privilèges objets

Compte utilisateur

- Chaque compte utilisateur de bases de données comporte:
  - Un nom de compte, un mot de passe
  - Un tablespace par défaut
  - Un tablespace temporaire
  - U n profil utilisateur
  - Une politique de gestion de mot de passe

Créer un compte utilisateur

```
CREATE USER nom_ut IDENTIFIED BY mdp
[DEFAULT TABLESPACE nom_def_TS]
[TEMPORARY TABLESPACE nom_tmp_TS]
[QUOTA {val|UNLIMITED} ON nom_TS]
[PROFILE nom_prof]
[PASSWORD EXPIRE]
[ACCOUNT {LOCK|UNLOCK}]
```

Authentification des utilisateurs

- Il existe 3 modes d'authentification pour les utilisateurs:
  - Password
  - External
  - Global
- Password (authentification par la base de données Oracle)
  - Ce mode crée chaque utilisateur avec un mot de passe associé qui doit être fourni lorsque l'utilisateur tente d'établir une connexion

Syntaxe : CREATE USER name IDENTIFIED BY password;

Authentification des utilisateurs

- Global
  - Permet l'identification des utilisateurs via la biométrie, les certificats x509, etc

  Syntaxe : CREATE USER name IDENTIFIED GLOBALLY;

- External (authentification par le système d'exploitation)
  - Le compte est géré par Oracle par contre l'administration des password et l'authentification seront assurées par un service externe (OS or a network service) :

  Syntaxe : CREATE USER name IDENTIFIED EXTERNALLY;

Comptes utilisateurs

- Modifier un compte utilisateur

Syntaxe : ALTER USER nom_ut [IDENTIFIED BY mdp]

- Supprimer un compte utilisateur

Syntaxe : DROP USER nom_ut [CASCADE] ;

- Les vues du dictionnaire de données qui fournissent des informations sur les utilisateurs ainsi que sur les quotas attribués aux utilisateurs sont **DBA_USERS** et **DBA_TS_QUOTAS.**

Les privilèges

- Un privilège est un droit avec lequel on peut permettre :
  - D'exécuter une tâche particulière de définition, de manipulation ou d'administration
  - D'accéder aux objets des autres utilisateurs et de pouvoir les manipuler
- Il existe deux types de privilèges utilisateur :
  - **Système** : permet aux utilisateurs d'effectuer des actions particulières dans la base de données
  - **Objet** : permet aux utilisateurs d'accéder à un objet spécifique et de le manipuler.

Les privilèges système

- Un privilège système est un privilège qui est relié avec le langage de définition de données (**LDD**) et le langage de contrôle de données (**LCD**).
- C'est un privilège qui donne le droit d'utiliser les requêtes **CREATE, ALTER,** et **DROP** sur les différents types d'objets relatifs à la base de données et aussi les requêtes **GRANT** et **REVOKE** qui permettent d'attribuer/retirer des privilèges.
- Les privilèges système sont présentés dans le dictionnaire de données dans la vue **DBA_SYS_PRIVS.**

Les privilèges systèmes

- Accorder un privilège

Syntaxe : GRANT priv1[,priv2…] TO {user1[,user2…]|PUBLIC} [WITH ADMIN OPTION]

- Un utilisateur peut attribuer un privilège si :

1. Le privilège lui a été attribué avec l'option WITH ADMIN
2. S'il a le privilège système GRANT ANY PRIVILEGE qui lui permet d'attribuer/retirer n'importe quel privilège.

Les privilèges objet

- Un privilège objet est le droit de manipuler un objet d'un schéma **(SELECT/ INSERT/ UPDATE/ DELETE)**, ou d'exécuter un programme PL/SQL **(EXECUTE)**.
- Le propriétaire d'un objet a naturellement tous les droits dessus. Il a aussi le droit de transférer ses privilèges à n'importe quel autre utilisateur.
- Les privilèges système sont présentés dans le dictionnaire de données dans la vue **DBA_TAB_PRIVS.**

Les privilèges objets

- Accorder un privilège

Syntaxe : GRANT priv1[,priv2…] TO {user1[,user2…]|PUBLIC} [WITH GRANT OPTION]

- Les utilisateurs qui peuvent attribuer/retirer un privilège objet à autrui sont

1. Le propriétaire de l'objet en question.
2. Un utilisateur ayant reçu le privilège avec l'option WITH GRANT OPTION.
3. Un utilisateur ayant le privilège système ANY OBJECT PRIVILEGE.

Retirer un privilège

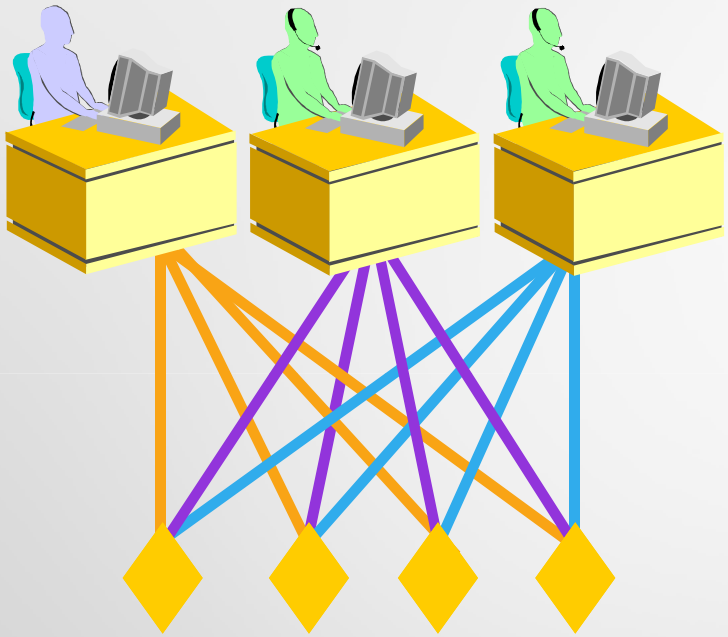
Syntaxe : REVOKE {priv1[,priv2…]|ALL PRIVILEGES} FROM {user1[,user2…]|PUBLIC}

- Revoke ne retire pas les privilèges en cascade.

```
SQL> CONNECT SYSTEM/MANAGER;
SQL> GRANT DROP ANY TABLE TO SCOTT WITH ADMIN OPTION;
SQL> CONNECT SCOTT/TIGER;
SQL> GRANT DROP ANY TABLE TO SMITH;
SQL> CONNECT SYSTEM/MANAGER;
SQL> REVOKE DROP ANY TABLE FROM SCOTT;
```

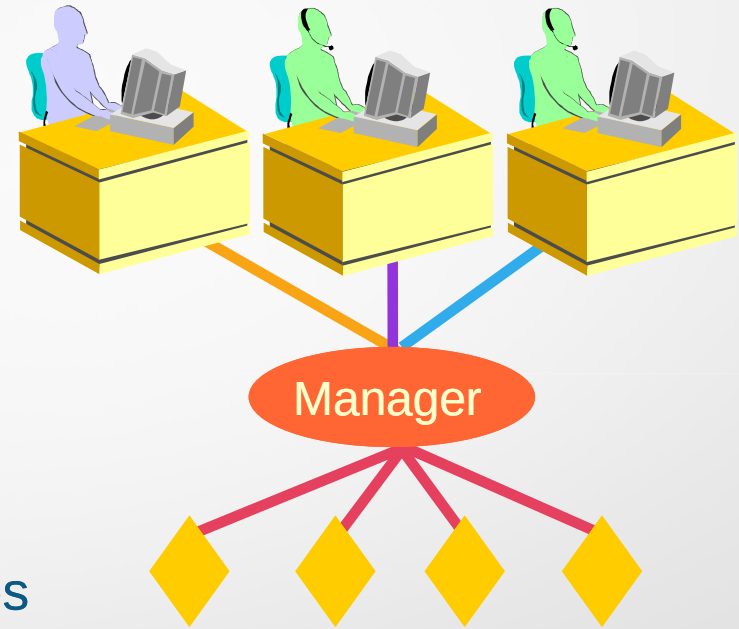
- le privilège système DROP ANY TABLE ne sera pas retiré de SMITH

Les Rôles



Users

Privileges



Affectation des privilèges sans rôles

Affectation des privilèges avec un rôle

Les Rôles

- Un rôle est un ensemble de privilèges qui est attribué à l'utilisateur.
- Lorsqu'on un rôle est attribué à un utilisateur, tous ses privilèges lui sont affectés.
- Un rôle peut être attribué à un autre rôle ;
- les privilèges du premier seront ajoutés à ceux du deuxième.
- Un utilisateur peut avoir plusieurs rôles.
- les informations sur les rôles sont accessibles dans les vues du dictionnaire de donnée **DBA_ROLES**

Avantages des rôles

- **Gestion plus facile des privilèges:** Plutôt que d'accorder le même ensemble de privilèges à plusieurs utilisateurs, on peut accorder les privilèges à un rôle, puis accorder ce rôle à chaque utilisateur
- **Gestion dynamique des privilèges:** Si les privilèges associés à un rôle sont modifiés, tous les utilisateurs auxquels ce rôle est accordé bénéficient automatiquement et immédiatement des privilèges modifiés
- **Disponibilité sélective des privilèges:** Les rôles peuvent être activés et désactivés afin d'activer ou de désactiver temporairement les privilèges. L'activation d'un rôle peut être utilisée pour vérifier que le rôle a été accordé à un utilisateur

Rôles prédéfinis

CONNECT	CREATE SESSION, CREATE TABLE, CREATE VIEW, CREATE SYNONYM, CREATE SEQUENCE, CREATE DATABASE LINK, CREATE CLUSTER, ALTER SESSION
RESOURCE	CREATE TABLE, CREATE PROCEDURE, CREATE SEQUENCE, CREATE TRIGGER, CREATE TYPE, CREATE CLUSTER, CREATE INDEXTYPE, CREATE OPERATOR
SCHEDULER_ ADMIN	CREATE ANY JOB, CREATE JOB, EXECUTE ANY CLASS, EXECUTE ANY PROGRAM, MANAGE SCHEDULER
DBA	La plupart des privilèges système et plusieurs autres rôles. Ce rôle ne doit pas être accordé aux utilisateurs qui ne sont pas administrateurs.
SELECT_ CATALOG_ROLE	Pas de privilèges système, mais plus de 1600 privilèges objet sur le dictionnaire de données.

Affecter des privilèges à des rôles et des rôles à des utilisateurs

- Création d'un rôle SUPERDBA avec mot de passe et intégrant le rôle DBA et des privilèges système

```
CREATE ROLE "SUPERDBA"  IDENTIFIED BY "sup";

GRANT ALTER ANY INDEXTYPE TO "SUPERDBA«

GRANT ALTER ANY  TABLE TO "SUPERDBA"

GRANT "DBA" TO "SUPERDBA";
```

- Affectation du rôle SUPERDBA à l'utilisateur STOCK

```
GRANT SUPERDBA TO STOCK;
```

Gestion des profils

- Principe :
  - Imposent un ensemble de limites concernant l'utilisation de la base de données et ses ressources
  - Gèrent le statut des comptes et les contraintes sur les mots de passe
  - Permettent le contrôle des ressources système (temps CPU, occupation réseau, etc…)
  - Un utilisateur est affecté à un seul profil

Gestion des profils

CREATE PROFILE < Nom de profil > LIMIT

[SESSIONS_PER_USER <nb max session>]

[CPU_PER_SESSION <val en seconde >]

[CPU_PER_CALL < val en seconde >]

[CONNECT_TIME <val en minute>]

[IDLE_TIME <val en minute>]

[FAILED_LOGIN_ATTEMPTS max_value]

[PASSWORD_LIFE_TIME max_value]

[PASSWORD_LOCK_TIME max_value]

[PASSWORD_GRACE_TIME max_value]

[PASSWORD_VERIFY_FUNCTION {function|null|DEFAULT}]

Limitation des ressources

- Condition : affecter la valeur TRUE au paramètre RESOURCE_LIMIT
- Paramètres de limitation :
  - Limiter la consommation du temps CPU → CPU_PER_SESSION
  - Limiter la consommation du temps par appel commande → CPU_PER_Call
  - Limiter le nombre de minutes pendant lequel un utilisateur peut être connecté avant d'être automatiquement déconnecté -> Connect_Time
  - Limiter le nombre de minutes pendant lequel une session utilisateur peut rester inactive avant d'être automatiquement déconnectée → Idle_Time
  - Limiter le nombre de session simultanées pouvant être crées à l'aide d'un compte utilisateur → SESSIONS_PER_USER

Exemple de création de profil de gestion de ressources

```
CREATE PROFILE profilename LIMIT
[SESSIONS_PER_USER max_value]
[CPU_PER_SESSION max_value]
[CONNECT_TIME max_value]
[IDLE_TIME max_value];
    max_value:={integer | UNLIMITED |
 DEFAULT};
```

Exemple:

```
CREATE PROFILE developer_profile LIMIT
  SESSIONS_PER_USER 2
  CPU_PER_SESSION 10000
  CONNECT_TIME 480
  IDLE_TIME 60;
```

Contraintes de gestion de mot de passe

- Appliquer un nombre de tentatives de connexion après lequel le compte est verrouillé → FAILED_LOGIN_ATTEMPTS
- Limiter le nombre de jours après lesquels un mot de passe expire → PASSWORD_LIFE_TIME
- Indiquer le nombre de jours pendant lesquels le compte reste verrouillé après un nombre de tentative de connexion

  → PASSWORD_LOCK_TIME

Contraintes de gestion de mot de passe

- Limiter le nombre de jours pendant lesquels un mot de passe peut être modifié → PASSWORD_GRACE_TIME
- Appliquer une fonction de complexité pour vérifier la saisie des mots de passe : cette fonction stockée doit être à la propriété de l'utilisateur SYS.Oracle fournit une fonction **utlpwdmg** par défaut → PASSWORD_VERIFY_FUNCTION

Affectation d'un profil à un utilisateur

- **ALTER USER** nom_user ***PROFILE*** nom_profil
- Les informations complètes concernant les profils sont consultables depuis la vue **DBA_PROFILES.**
- **Modifier un profil**

  ALTER PROFILE profilename LIMIT

  [SESSIONS_PER_USER max_value]

  [CPU_PER_SESSION max_value] …,
- **Supprimer un profil**

  DROP PROFILE profile_name