Unidad 2

Control del acceso a la base de datos. DCL

mediante Oracle 11g

Apuntes realizados para la asignatura de FP Grado Superior:

Administración de Bases de Datos

del ciclo Administración de Sistemas Informáticos en Red

Autor: Jorge Sánchez Asenjo (www.jorgesanchez.net)

Versión del documento: 2.11, Año 2012

índice

(2) control de acceso a la base de datos. DCL

(2.1) introducción

Todo acceso a una base de datos requiere conectar mediante un usuario y contraseña. Dicho usuario dará derecho a utilizar ciertos objetos de la base de datos, pero se puede restringir el uso de otros.

A los usuarios se les asigna una serie de privilegios que son los que dan permiso de uso a ciertos objetos. Para organizarse mejor la mayoría de Sistemas Gestores de Bases de Datos permiten agrupar permisos que normalmente se aplican conjuntamente en estructuras llamadas perfiles y roles, que en definitiva son un conjunto de permisos.

Por ello cuando un usuario conecta debe probar que es quien dice ser (normalmente mediante una contraseña), es decir se autentifica. Por otro lado esta autentificación dará lugar a unos privilegios (unos derechos) y unas restricciones.

Todo lo que se explica en este tema se refiere a la gestión de usuarios en la base de datos Oracle 11g.

(2.2) cuentas y permisos administrativos

(2.2.1) cuentas administrativas

Durante la instalación de Oracle se instalan dos cuentas administrativas y otras dos con permisos especiales para tareas de optimización y monitorización de la base de datos:

- SYS. Inicialmente posee la contraseña CHANGE_ON_INSTALL que, lógicamente hay que cambiar inmediatamente en la istalación. SYS toma rol de DBA y en su esquema se crea el diccionario de datos, por lo que no conviene de ninguna manera crear otro tipo de elementos en su esquema.
- SYSTEM. Posee también el rol DBA y se crea durante la instalación. Como antes, la contraseña MANAGER que tiene por defecto se debería cambiar en la

instalación. En su esquema se suelen crear tablas y vistas administrativas (pero no se deberían crear otro tipo de tablas).

- **SYSMAN**. Usado para realizar tareas administrativas con la aplicación **Enterprise Manager**.
- **DBSMNP.** Monitoriza Enterprise Manager.

(2.2.2) privilegios administrativos

Oracle posee dos privilegios de sistema asociados a tareas administrativas, son:

- **SYSDBA.** Con capacidad de parar e iniciar (instrucciones **SHUTDOWN** y **STARTUP**) la instancia de base de datos; modificar la base de datos (**ALTER DATABASE**), crear y borrar bases de datos (**CREATE** y **DROP DATABASE**), Crear el archivo de parámetros (**CREATE SPFILE**), cambiar el modo de archivado de la base de datos, recuperar la base de datos y además incluye el privilegio de sistema **RESTRICTED SESSION**. En la práctica es usar el usuario **SYS**.
- **SYSOPER.** Permite lo mismo que el anterior salvo: crear y borrar la base de datos y recuperar en todas las formas la base de datos (hay modos de recuperación que requieren el privilegio anterior).

La vista **V$PWFILE_USERS** nos permite examinar a los usuarios administrativos.

(2.3) características de los usuarios de Oracle

A los usuarios de Oracle se les puede asignar la configuración referida a:

- **Nombre de usuario**. No puede repetirse y como máximo debe tener 30 caracteres que sólo podrán contener letras del alfabeto inglés,números, el signo dólar y el signo de guión bajo (_)
- **Configuración física**. Se refiere al espacio asociado al usuario para almacenar sus datos (lo que Oracle llama **tablespace**) y la cuota (límite de almacenamiento) que se le asigna.
- **Perfil asociado**. El perfil del usuario indica los recursos y configuración que tomará el usuario al sistema
- **Privilegios y roles**. Permiten especificar las acciones que se le permiten realizar al usuario.
- **Estado de la cuenta de usuario**:
  - **Abierta**. El usuario puede conectar y realizar sus acciones habituales
  - **Bloqueada**. EL usuario no podrá conectar mientars siga en estado bloqueado. El bloqueo lo realiza el DBA:

    ```
    ALTER USER usuario ACCOUNT LOCK
    ```

  - **Expirada**. La cuenta agotó el tiempo máximo asignado a ella. Para salir de este estado, el usuario/a debe resetear su contraseña de usuario.
  - **Expirada y bloqueada**.
  - **Expirada en periodo de gracia**. Está en los últimos momentos de uso antes de pasar a estado de expirada

(2.4) autentificación

La autentificación define la forma en la que el usuario verifica quién es. Hay métodos de autentificación más seguros que otros. Asegurar la autentificación implicaría asegurar el medio la comunicación entre usuario y base de datos con protocolos de cifrado. Por otro lado hay que proteger especialmente a los usuarios administradores.

(2.4.1) autentificación por el sistema operativo

Se permite el uso sólo en usuarios con privilegios administrativos. En el sistema operativo en el que se instale Oracle se crean dos **grupos** de usuarios relacionados con los dos privilegios de sistema **SYDBA** y **SYSOPER**. En Windows se llaman **ORA_DBA** y **ORA_OPER** respectivamente, en Linux simplemente **dba** y **oper**.

Los usuarios de esos grupos conectarían mediante **CONNECT / AS SYSDBA** o **CONNECT / AS SYSOPER**.

Otra posibilidad es conectar con:

```
CONNECT /@servicioRed  AS SYSDBA
```

En este caso usamos los privilegios del sistema operativo para conectar con una base de datos remota cuyo nombre de servicio de red se indique. Esta forma sólo vale para máquinas dentro de un dominio Windows.

(2.4.2) autentificación por archivo de contraseñas

Se usa en los mismos casos que la anterior. Cuando no se considera que el Sistema Operativo sea muy seguro, se utiliza como opción. Para usar esta forma de autentificación los usuarios de tipo SYSDBA o SYSOPER indican su nombre de usuario y contraseña al conectar (opcionalmente indican el host al que se desean conectar) esos datos se contrastarán con los del archivo de contraseñas utilizado. Esta forma (y la anterior) permite conectar la base de datos aunque no esté montada todavía la base de datos.

La utilidad ORAPWD permite crear, si no existe el archivo de contraseñas:

```
ORAPWD FILE=ruta    [ENTRIES=n [FORCE=y|n[IGNORECASE=y|n]]]
```

Funcionamiento:

- **FILE**. Permite indicar el nombre del archivo de contraseñas
- **ENTRIES**. Indica el máximo número de contraseñas que admitirá el archivo
- **FORCE**. En caso de darle el valor **y** sobrescribe las contraseñas cuando asignemos una nueva a un usuario administrativo
- **IGNORECASE**. No tiene en cuenta mayúsculas ni minúsculas en las contraseñas.

Por otra lado el parámetro de sistema **REMOTE_LOGIN_PASSWORDFILE** (modificable con **ALTER SYSTEM SET**…), visto en el tema anterior) permite indicar la forma en la que funciona el archivo de contraseñas. Valores posibles:

- **NONE**. No permite usar el archivo de contraseñas, las conexiones de usuarios con privilegios administrativos tendrán que usar otros métodos.

- EXCLUSIVE. El archivo de contraseñas se usa sólo en la instancia actual.
- SHARED: Se comparte el archivo de contraseñas entre varias instancias de tipo Real Application Cluster de Oracle (para bases de datos distribuidas). En este caso no se pueden cambiar las contraseñas de los usuarios administrativos.

(2.4.3) autentificación por contraseña

En este caso los usuarios son autentificados mediante una contraseña que se contrastará en el diccionario de datos que es donde se almacenan estas contraseñas. Es el método habitual de autentificación para usuarios no administrativos. Está configuración requiere la base de datos montada y abierta (al tener que usar el diccionario de datos).

La contraseña se pasa encriptada desde el ordenador cliente al servidor mediante el algoritmo AES.

(2.4.4) autentificación externa

Oracle delega la autentificación a un servicio externo que se asociará a Oracle. Ejemplos de servicios externos son Kerberos o RADIUS, este último sólo disponible en Windows. Requiere el uso de las mejoras de seguridad avanzada de Oracle.

(2.4.5) autentificación global

Se trata de utilizar un servicio LDAP para realizar la autentificación. Oracle dispone de un servicio LDAP global integrado en Oracle Applications (plataforma de Oracle para le creación de aplicaciones) que se llama Oracle Internet Directory.

Si los usuarios sólo están dados de alta en el directorio externo, usarán todos la misma cuenta de Oracle; para independizarles se requiere darles de alta en ambos servicios (Oracle y el Oracle Internet Directory).

(2.5) control de usuarios en Oracle

(2.5.1) creación de usuarios en Oracle

La sentencia de creación de usuarios (que es estándar) es:

```
CREATE USER nombre IDENTIFIED BY 'contraseña' [OPCIONES]
```

Es una sentencia estándar, a la que se le pueden añadir múltiples cláusulas.

```
CREATE USER nombre {IDENTIFIED BY 'contraseña'  |
                    EXTERNALLY |
                    GLOBALLY AS nombreGlobal}
[DEFAULT TABLESPACE tableSpacePorDefecto]
[TEMPORARY TABLESPACE tableSpacetTemporal]
[QUOTA {cantidad [K|M] | UNLIMITED} ON tablespace
  [QUOTA {cantidad [K|M] | UNLIMITED} ON tablespace [...]]
]
[PASSWORD EXPIRE]
[ACCOUNT {UNLOCK|LOCK}];
[PROFILE {perfil | DEFAULT}]
```

Sólo la primera línea es obligatoria, el resto posee opciones por defecto que se aplican si no se especifica cada apartado (no hace falta especificar todos, sólo las líneas que nos interesen). Ejemplo:

```
CREATE USER jsanchez IDENTIFIED BY 'Caracola'
DEFAULT TABLESPACE 'Usuarios'
QUOTA 15M ON 'Usuarios' //Se dan 15MBytes de espacio en el tablespace
ACCOUNT LOCK; //La cuenta estará bloqueada
```

(2.5.2) modificación de usuarios

Cada parámetro indicado anteriormente se puede modificar mediante la instrucción ALTER USER que se utiliza igual que CREATE USER. Ejemplo:

```
ALTER USER jsanchez QUOTA UNLIMITED ON usuarios
```

(2.5.3) borrado de usuarios

Se realiza mediante:

```
DROP USER usuario [CASCADE]
```

La opción CASCADE elimina los objetos del esquema del usuario antes de eliminar al propio usuario. Es obligatorio si el esquema contiene objetos.´

(2.5.4) consultar usuarios

La vista administrativa DBA_USERS muestra la lista y configuración de todos los usuarios del sistema. Para observar la estructura de la vista, siempre es conveniente usar DESCRIBE DBA_USERS.

(2.6) control de privilegios en Oracle

Los privilegios son permisos que damos a los usuarios para que puedan realizar ciertas operaciones con la base de datos. En Oracle hay más de cien posibles privilegios. Se dividen en:

- **Privilegios de sistema**. Son permisos para modificar el funcionamiento de la base de datos. Son cambios, en definitiva, que afectan a todos los usuarios y usuarias.
- **Privilegios de objeto**. Son permisos que se aplican a un objeto concreto de la base de datos.

(2.6.1) privilegios de sistema

Se comentan algunos de los privilegios de sistema más importantes

Privilegio	Significado
CREATE SESSION	Permite al usuario conectar con la base de datos
RESTRICTED SESSION	Permite al usuario establecer sesión con la base de datos en caso de que la base de datos esté en modo restringido mediante la instrucción: ALTER SYSTEM ENABLE RESTRICTED SESSION Sólo los usuarios con este privilegio puede conectar con la base de datos si ésta se encuentra en este modo.
ALTER DATABASE	Permite modificar la estructura de la base de datos
ALTER SYSTEM	Permite modificar los parámetros y variables del sistema
CREATE TABLE	Permite crear tablas. Incluye la posibilidad de borrarlas.
GRANT ANY OBJECT PRIVILEGE	Permite conceder privilegios sobre objetos que no son del usuario (pertenecen a otros usuarios) a terceros usuarios.
CREATE ANY TABLE	Permite crear tablas en otros esquemas de usuario
DROP ANY TABLE	Permite borrar tablas de otros usuarios
SELECT ANY TABLE	Permite seleccionar datos en tablas de otros usuarios
INSERT ANY TABLE	Permite añadir datos en tablas de otros usuarios
UPDATE ANY TABLE	Permite eliminar datos en tablas de otros usuarios

Privilegio	Significado
DELETE ANY TABLE	Permite eliminar datos en tablas de otros usuarios

En la tabla anterior se ha hecho hincapié en los privilegios referidos a las tablas, para otros objetos el funcionamiento es similar: igual que hay **CREATE TABLE**, se puede usar **CREATE VIEW** para las vistas o **INDEX**, **TRIGGER**, **PROCEDURE**, **SEQUENCE**, **SYNONYM**, **TYPE**,... y de esa forma podemos conceder privilegio de creación de otros objetos. Lo mismo con el resto de operaciones

Hay dos privilegios especiales que permiten conceder nivel de DBA, son: **SYSDBA** y **SYSOPER**. Se han comentado anteriormente.

La lista completa de privilegios es:

Privilegio	Permite
Sesiones	
CREATE SESSION	Conectar a la base de datos
ALTER SESSION	Modificar el funcionamiento de la sesión
ALTER RESOURCE COST	Modifica los parámetros de cálculo de coste de la sesión
RESTRICTED SESSION	Conectar aunque la base de datos se haya iniciado en modo restringido
Base de datos y sistema	
ALTER DATABASE	Modificar la base de datos (privilegio de gran capacidad administrativa)
ALTER SYSTEM	Modificar los parámetros del sistema
AUDIT SYSTEM	Auditar la base de datos
Usuarios, roles, privilegios y perfiles	
CREATE USER	Crear usuarios pudiendo indicar tablespace por defecto, cuotas y perfiles
ALTER USER	Modificar al usuario. Permite cambiar la contraseña y modo de autentificación, tablespace por defecto, cuota de uso de disco, roles y el perfil del usuario
DROP USER	Borrar usuario
CREATE PROFILE	Crear perfiles
ALTER PROFILE	Modificar perfiles
DROP PROFILE	Borrar perfiles
CREATE ROLE	Crear roles
ALTER ANY ROLE	Modificar roles
GRANT ANY ROLE	Conceder roles
GRANT ANY PRIVILEGE	Conceder privilegios de objeto
GRANT ANY SYSTEM PRIVILEGE	Conceder privilegios de sistema
Directorios	
CREATE ANY DIRECTORY	Crear directorios
DROP ANY DIRECTORY	Borrar directorios

Privilegio	Permite
Tablespaces (espacios de tabla)	
CREATE TABLESPACES	Crear tablespaces
ALTER TABLESPACE	Modificar tablespaces
DROP TABLESPACE	Borrar tablespaces
MANAGE TABLESPACE	Administrar el espacio de tablas para poder hacer copia de seguridad o simplemente quedar online u offline el tablespace
UNLIMITED TABLESPACE	Usa cuota ilimitada al escribir en cualquier tablespace. Este privilegio elimina las cuotas establecidas sobre el usuario, si las hubiera.
Tablas	
CREATE TABLE	Crear tablas en el esquema del usuario, incluye insertar, modificar y eliminar datos de la misma; así como eliminar la propia tabla
ALTER ANY TABLE	Modificar tablas de cualquier usuario
BACKUP ANY TABLE	Utilizar la utilidad **Export** para copiar datos de otros esquemas.
CREATE ANY TABLE	Crear tablas en cualquier esquema
DELETE ANY TABLE	Borrar filas de tablas en cualquier esquema
DROP ANY TABLE	Borrar tablas en cualquier esquema
INSERT ANY TABLE	Añadir datos a cualquier tabla
SELECT ANY TABLE	Seleccionar datos de tablas en cualquier esquema
UPDATE ANY TABLE	Modificar datos de tablas de cualquier esquema
LOCK ANY TABLE	Bloquear tablas, vistas e instantáneas en cualquier esquema
FLASHBACK ANY TABLE	Realizar acción de flashback en tablas, vistas e instantáneas en cualquier esquema
Vistas	
CREATE VIEW	Crear vistas en el esquema del usuario
CREATE ANY VIEW	Crear vistas en cualquier esquema
DROP ANY VIEW	Borrar cualquier vista en cualquier esquema
UNDER ANY VIEW	Crear subvistas
Instantáneas (Snapshots o vistas materializadas)	
CREATE MATERIALIZED VIEW	Crear vistas materializadas (instantáneas)
CREATE ANY MATERIALIZED VIEW	Crear vistas materializadas (instantáneas) en cualquier esquema
ALTER ANY MATERIALIZED VIEW	Modificar vistas materializadas (instantáneas) en cualquier esquema
DROP ANY MATERIALIZED VIEW	Borrar vistas materializadas (instantáneas) en cualquier esquema
GLOBAL QUERY REWRITE	Permite realizar operaciones de lectura escritura en instantáneas que usan tablas de otros esquemas
CREATE SNAPSHOT	Crear instantáneas (obsoleto)

Privilegio	Permite
ALTER ANY SNAPSHOT	Modificar instantáneas de cualquier usuario (obsoleto)
CREATE ANY SNAPSHOT	Crear instantáneas a cualquier usuario (obsoleto)
DROP ANY SNAPSHOT	Borrar instantáneas (obsoleto)
PL/SQL	
CREATE PROCEDURE	Crear procedimientos y funciones PL/SQL
ALTER ANY PROCEDURE	Modificar procedimientos y funciones de cualquier usuario
CREATE ANY PROCEDURE	Crear funciones y procedimientos en cualquier esquema
DROP ANY PROCEDURE	Borrar cualquier procedimiento en cualquier esquema
EXECUTE ANY PROCEDURE	Ejecutar cualquier procedimiento en cualquier esquema
CREATE TRIGGER	Crear triggers
ALTER ANY TRIGGER	Modificar triggers de cualquier usuario
CREATE ANY TRIGGER	Crear triggers en cualquier esquema
DROP ANY TRIGGER	Borrar triggers de cualquier esquema
ADMINISTER DATABASE TRIGGER	Crear triggers de sistema (requiere además el privilegio CREATE TRIGGER)
CREATE LIBRARY	Crear librerías de procedimientos y funciones en el esquema de usuario
CREATE ANY LIBRARY	Crear librerías de procedimientos y funciones en cualquier esquema
DROP ANY TRIGGER	Borrar cualquier trigger
DROP LIBRARY	Borrar librería de procedimientos y funciones en el esquema de usuario
DROP ANY LIBRARY	Borrar librerías de procedimientos y funciones en cualquier esquema
EXECUTE ANY LIBRARY	Ejecutar cualquier librería
Tipos de datos	
CREATE TYPE	Crear tipos de datos personales
ALTER ANY TYPE	Modificar tipos de datos personales en cualquier usuario
CREATE ANY TYPE	Crear tipos de datos en cualquier esquema
DROP ANY TYPE	Borrar tipos de datos de cualquier esquema
EXECUTE ANY TYPE	Permite invocar a tipos de datos personales presentes en cualquier esquema
Índices	
ALTER ANY INDEX	Modificar índices de la base de datos (incluye modificar claves primarias, secundarias,...)
CREATE ANY INDEX	Crear índices en cualquier esquema
DROP ANY INDEX	Borrar índices en cualquier esquema
Secuencias y sinónimos	
ALTER ANY SEQUENCE	Modificar secuencias de cualquier usuario
CREATE ANY SEQUENCE	Crear secuencias en cualquier esquema
CREATE ANY SYNONYM	Crear sinónimos en cualquier esquema

Privilegio	Permite
CREATE SEQUENCE	Crear secuencias
CREATE SYNONYM	Crear sinónimos
CREATE PUBLIC SYNONYM	Crear sinónimos públicos
DROP PUBLIC SYNONYM	Borrar sinónimos públicos
CREATE ANY SEQUENCE	Crear secuencias en cualquier esquema
DROP ANY SEQUENCE	Borrar secuencias en cualquier esquema
DROP ANY SYNONYM	Borrar sinónimos en cualquier esquema
SELECT ANY SEQUENCE	Seleccionar cualquier secuencia de cualquier esquema
Clusters	
CREATE CLUSTER	Crea y modifica clusters en el esquema actual
ALTER ANY CLUSTER	Modificar clusters
CREATE ANY CLUSTER	Crear clusters en cualquier esquema
DROP ANY CLUSTER	Borrar cualquier cluster
Segmentos de rollback	
CREATE ROLLBACK SEGMENT	Crear segmentos de rollback
ALTER ROLLBACK SEGMENT	Modificar segmentos de rollback
DROP ROLLBACK SEGMENT	Borrar segmento de rollback
Enlaces a base de datos	
CREATE DATABASE LINK	Crear enlaces privados a bases de datos en el esquema del usuario
CREATE PUBLIC DATABASE LINK	Crear enlaces públicos a bases de datos
CREATE DATABASE LINK	Modificar enlaces privados a bases de datos
CREATE PUBLIC DATABASE LINK	Modificar enlaces públicos a bases de datos
DROP PUBLIC DATABASE LINK	Borrar enlaces públicos a bases de datos
Programación de tareas	
CREATE JOB	Crear trabajo planificado en el esquema actual
CREATE ANY JOB	Crea, modifica y elimina tareas, programas y credenciales de cualquier esquema (excepto SYS). Esto permite ejecutar código en cualquier esquema de cualquier usuario.
CREATE EXTERNAL JOB	Crear un trabajo en el esquema de usuario procedente del planificador de tareas del sistema operativo
EXECUTE ANY PROGRAM	Ejecutar cualquier programa presente en un trabajo planificado del esquema de usuario.
EXECUTE ANY CLASS	Asignar cualquier clase a un trbajo en el esquema de usuario.
MANAGE SCHEDULER	Administrar el planificador de tareas,
Varios	
ANALYZE ANY	Analizar cualquier tabla, clúster o índice en cualquier esquema.

Privilegio	Permite
ANALYZE ANY DICTIONARY	Analizar cualquier elemento del diccionario de datos
SELECT ANY DICTIONARY	Realizar SELECT sobre las vistas del diccionario de datos
AUDIT ANY	Auditar a cualquier objeto de la base de datos
BECOME USER	Convertirse en otro usuario al utilizar algunas de las utilidades de Oracle
COMMENT ANY TABLE	Realizar comentarios sobre tablas, columnas y vistas en cualquier esquema de la base de datos
SELECT ANY TRANSACTION	Seleccionar los datos de la vista FLASHBACK_TRANSACTION_QUERY que controla el proceso de la actual operación flashback.
FORCE ANY TRANSACTION	Forzar aceptar (COMMIT) las transacciones en duda en un sistema distribuido de bases de datos en cualquier conexión
FORCE TRANSACTION	Forzar aceptar (COMMIT) la transacción actual en caso de duda.
SYSDBA	Privilegio general de administrador
SYSOPER	Privilegio general de administrador (más bajo que el anterior)
FLASHBACK ARCHIVE ADMINISTER	Crea, elimina o modifica cualquier archivo de flashback
DEBUG CONNECT SESSION	Conectar la sesión a un depurador
DEBUG ANY PROCEDURE	Conectar procedimientos, funciones y/o código Java a un depurador

(2.6.2) conceder privilegios

Se usa con la instrucción GRANT que funciona así:

```
GRANT privilegio1 [,privilegio2[,...]] TO usuario
[WITH ADMIN OPTION];
```

La opción WITH GRANT OPTION permite que el usuario al que se le concede el privilegio puede conceder dicho privilegio a otros usuarios. Es, por tanto, una opción a utilizar con cautela.

Ejemplo:

```
GRANT    CREATE SESSION, ALTER SESSION, CREATE TABLE,
         CREATE VIEW, CREATE SYNONYM, CREATE SEQUENCE,
         CREATE TRIGGER, CREATE PROCEDURE, CREATE TYPE
TO jsanchez;
```

(2.6.3) revocar

Se realiza con la instrucción REVOKE que funciona de esta forma:

```
REVOKE privilegio1 [,privilegio2 [,...]]  FROM usuario;
```

Al revocar los privilegios, las acciones llevadas a cabo con ellos no se anulan.

(2.6.4) privilegios de objeto

Se trata de privilegios que se colocan a un objeto para dar permiso de uso a un usuario.
Sintaxis:

```
GRANT {privilegio [(listaColumnas)] [,privilegio [(listaColumnas)] [,...]] |
ALL [PRIVILEGES]}
ON [esquema.]objeto
TO {usuario | rol | PUBLIC} [,{usuario | rol | PUBLIC} [,...]]
[WITH GRANT OPTION]
```

La opción ALL concede todos los privilegios posibles sobre el objeto. Se pueden asignar varios privilegios a la vez y también varios posibles usuarios. La opción WITH GRANT OPTION permite al usuario al que se le conceden los privilegios, que pueda, a su vez, concederlos a otro.

Ejemplo de uso de GRANT con privilegios de objeto:

```
GRANT UPDATE, INSERT ON jsanchez.personas TO anozal;
```

Los privilegios posibles están en la siguiente tabla:

Privilegio	Aplicable a
SELECT	Tablas, vistas, instantáneas
INSERT	Tablas, vistas
UPDATE	Tablas, vistas
DELETE	Tablas, vistas
ALTER	Tablas, secuencias
EXECUTE	Procedimientos, funciones, paquetes, sinónimos, programas en directorios
INDEX	Tablas (para crear índices en la misma)
REFERENCES	Tablas (para crear claves secundarias, FOREIGN KEY)
UNDER	Vistas, para crear subvistas
DEBUG	Depurar procedimientos y funciones mediante programa externo
ON COMMIT REFRESH	Actualizar la vista materializada (o instantánea) al realizar un COMMIT
QUERY REWRITE	Escribir en la vista materializada (o instantánea)
READ	Directorios

Privilegio	Aplicable a
WRITE	Directorios
FLASHBACK ARCHIVE	Archivos de datos flashback (activar o desactivar)

(2.6.5) quitar privilegios de objeto

Sintaxis:

```
REVOKE {privilegio1 [,privilegio2] [,...]] |
ALL [PRIVILEGES]}
ON [esquema.]objeto
FROM {usuario | rol | PUBLIC} [,{usuario | rol | PUBLIC} [,...]]
[CASCADE CONSTRAINTS]
```

CASCADE CONSTRAINTS elimina cualquier restricción que impida el borrado del privilegio.

(2.6.6) mostrar información sobre privilegios

Las vistas que permiten mostrar información sobre privilegios son:

Vista	Significado
DBA_SYS_PRIVS	Privilegios de sistema asignados a usuarios y roles
DBA_TAB_PRIVS	Lista de todos los privilegios de todos los objetos de la base de datos
DBA_COL_PRIVS	Lista de todos los privilegios aplicados a columnas de la base de datos
SESSION_PRIVS	Privilegios en activo para el usuario y sesión actuales

(2.7) administración de roles en Oracle

Los roles son privilegios aglutinados sobre un mismo nombre, bajo la idea de que ese conjunto denote un uso habitual sobre la base de datos. Gracias a los roles se facilita la asignación de privilegios a los usuarios. Un usuario puede tener asignados varios roles y viceversa.

(2.7.1) creación de roles

Los roles se crean usando esta sintaxis

```
CREATE ROLE rol [NOT IDENTIFIED |
   IDENTIFIED {BY password | EXTERNALLY |
   GLOBALLY | USING package}];
```

La opción IDENTIFIED funciona igual que las formas de identificar un usuario, salvo la opción PACKAGE que hace que el rol sólo se pueda utilizar para el paquete de aplicaciones indicado. Por defecto un ROL no requiere identificación.

La instrucción ALTER ROLE permite modificar la configuración del rol (tiene las mismas opciones que CREATE ROLE)

(2.7.2) asignar y retirar privilegios a roles

Se realiza con la instrucción GRANT. A los roles se les asignan privilegios igual que a los usuarios, pueden ser de sistema y/o de objeto.

Lógicamente se eliminan mediante REVOKE.

(2.7.3) asignar roles

Se pueden asignar usuarios a un usuario e incluso a otro rol. La sintaxis es:

```
GRANT rol1 [,rol2 [,...]]
TO {usuario|rol|PUBLIC [,{usuario|rol|PUBLIC} [,...] }
[WITH ADMIN OPTION]
```

Al igual que en las instrucciones anteriores, PUBLIC asigna el rol a todos los usuarios y WITH ADMIN OPTION permite al usuario al que se le concede el rol, conceder él dicho rol a otros usuarios/as.

(2.7.4) roles predefinidos

Oracle dispone de una serie de roles predefinidos que se pueden asignar a los usuarios. Hay más de cincuenta roles predefinidos. Los clásicos son:

rol	significado
CONNECT	Permite crear sesiones. Se mantiene por compatibilidad
RESOURCE	Permite crear tablas y código PL/SQL del tipo que sea. Se mantiene por compatibilidad
DBA	Permite casi todo, excepto manejar la instancia de la base de datos

(2.7.5) activar y desactivar roles

Cuando un usuario inicia sesión, por defecto todos los roles salvo el rol por defecto están desactivados.

La activación de un rol se realiza mediante SET ROLE que se encarga de desactivar y activar roles. Su sintaxis:

```
SET ROLE
  { rol1 [IDENTIFIED BY contraseña]
   [,rol2 [IDENTIFIED BY contraseña] [,...]]
   | ALL [EXCEPT rol1 [,rol2 [,...]]]
   | NONE};
```

Las posibilidades son:

- Indicar una lista de roles que serán los que se activen (se usa cuando se habían desactivado)
- Indicar ALL para activar todos los roles, excepto aquellos que se indiquen en la cláusula EXCEPT que quedarán sin activar.
- NONE desactiva todos los roles, sólo quedará activadol el rol marcado por defecto y los privilegios individuales marcados explícitamente.

(2.7.6) asignar a un usuario un rol por defecto

La instrucción que asigna un rol por defecto es:

```
ALTER USER usuario
DEFAULT ROLE {rol1 [,rol2 [,...]| ALL [EXCEPT rol1 [,rol2[,...]] |NONE ]};
```

Funciona de forma similar a SET ROL, pero sirve para colocar un rol por

(2.7.7) borrar roles

Lo hace la instrucción DROP ROLE, seguida del rol a borrar. Desde ese momento a los usuarios a los que se habían asignado el rol se les revoca.

(2.7.8) información sobre roles

Existen varias vistas para examinar los roles.

Vista	Significado
DBA_ROLES	Muestra todos los roles de la base de datos
DBA_ROLES_PRIVS	Roles asignados a los usuarios
ROLE_ROLE_PRIVS	Roles asignados a otros roles
DBA_SYS_PRIVS	Privilegios de sistema asignados a usuarios y roles
ROLE_SYS_PRIVS	Privilegios de sistema asignados a roles
ROLE_TAB_PRIVS	Privilegios de objeto concedidos a roles
SESSION_ROLES	Roles en activo para el usuario actual

(2.8) administración de perfiles de Oracle

Los perfiles permiten limitar los recursos que los usuarios usan de la base de datos. Hay un perfil llamado DEFAULT que se aplica automáticamente a todos los usuarios y que les da recursos ilimitados sobre la base de datos. Para limitar el número de recursos en principio se debe de activar a TRUE la variable de sistema RESOURCE_LIMIT (que por defecto está a FALSE). Esto se hace:

```
ALTER SYSTEM SET RESOURCE_LIMIT=TRUE;
```

En realidad hay dos tipos de parámetros de los perfiles:

- **Manejo de contraseñas**, los posibles cambios respecto a ese aspecto son:

Variable de perfil	Significado
FAILED_LOGIN_ATTEMPTS	Número consecutivo de errores en las contraseñas antes de bloquear la cuenta. Por defecto son 10
PASSWORD_LOCK_TIME	Número de días hasta que se bloquea una cuenta si se supera el límite de intentos al meter una contraseña. Por defecto es uno
PASSWORD_LIFE_TIME	Números de días que tiene vigencia una contraseña. Por defecto es 180
PASSWORD_GRACE_TIME	Días que la contraseña se la concede un periodo extra de gracia tras consumir su tiempo de vida. Por defecto es 7
PASSWORD_REUSE_TIME	Número de días que una contraseña puede ser reutilizada
PASSWORD_VERIFY_FUNCTION	Función a la que se invoca cuando se modifica una contraseña con el fin de verificar su validez en base a las reglas de complejidad que deseemos

- **Manejo de recursos.**

Variable de perfil	Significado
SESSIONS_PER_USER	Número de conexiones de usuario concurrentes que se permiten.
CPU_PER_SESSION	Límite de tiempo (en centésimas de segundo) que se permite a un usuario utilizar la CPU antes de ser echado del sistema. De esa forma se evitan peligros de rendimiento
CPU_PER_CALL	Como la anterior pero referida a cada proceso
PRIVATE_SGA	Para conexiones en instalaciones de servidor compartido, número de KB que puede consumir cada sesión en la zona de memoria compartida (SGA)
CONNECT_TIME	Minutos como máximo que se permite a una sesión
IDLE_TIME	Minutos máximos de inactividad de una sesión
LOGICAL_READS_PER_SESSION	Máximo número de bloques leídos en una sesión
LOGICAL_READS_PER_CALL	Máximo número de bloques leídos por un proceso
COMPOSITE_LIMIT	Máximo número de recursos consumidos por una sesión. Es la media ponderada de varios parámetros anteriores

(2.8.1) crear perfiles

Sintaxis:

```
CREATE PROFILE perfil LIMIT parámetros
```

Los parámetros son los explicados anteriormente a los que se les indica un valor, o bien la palabra DEFAULT (significa que el parámetro toma su valor por defecto) o bien UNLIMITED para indicar que no tienen límite. Ejemplo:

```
CREATE PROFILE programador LIMIT
        SESSIONS_PER_USER UNLIMITED
        CPU_PER_SESSION UNLIMITED
        IDLE_TIME 15
        CONNECT_TIME 150
        FAILED_LOGIN_ATTEMPTS 5
        PASSWORD_LOCK_TIME 2;
```

(2.8.2) modificar perfiles

La instrucción ALTER PROFILE funciona igual que CREATE PROFILE y es la encargada de hacer modificaciones a un perfil creado. Permite hacer modificaciones al perfil que se usa por defecto (DEFAULT).

(2.8.3) borrar perfil

En este caso es DROP PROFILE seguida del nombre del perfil a eliminar. Se puede usar la palabra CASCADE para eliminar todas las restricciones que impidan crear el perfil. Sintaxis:

```
DROP PROFILE perfiL [CASCADE]
```

(2.8.4) asignar un perfil a un usuario

Cada usuario tiene un solo perfil. La instrucción de creación de usuarios ya dispone de apartado para indicar el perfil que se asigna. Pero si lo deseamos hacer después disponemos de la instrucción ALTER USER con la que podemos indicar el perfil para el usuario. Ejemplo:

```
ALTER USER jsanchez PROFILE programador;
```

(2.9) APÉNDICE: usuarios y privilegios en MySQL

(2.9.1) cuentas de usuario en MySQL

En MySQL el nombre de un usuario está compuesto por el nombre seguido del signo @ y después el ordenador desde el que dicho usuario se conecta, porque se asume que no es lo mismo el usuario ***pepe@192.168.1.35*** que ***pepe@192.168.1.36***, es decir hay diferentes usuarios de nombre ***pepe*** y tendrán por tanto diferentes privilegios según de qué ***pepe*** hablemos en base a la máquina o la red desde la que se conectan.

A partir de esta idea, cuando un usuario se conecta primero se comprueba si tiene permiso para hacerlo (suponiendo que la contraseña sea correcta). Después cada operación que intenta realizar será controlada para saber si se permite o no.

La tabla **mysql.user**, es decir: tabla **user** de la base de datos **mysql** que contiene la información sobre los usuarios de mysql. En ella se observa una de las particularidades de MySQL, los usuarios usan un nombre seguido del host. De esa forma dos usuarios pueden parecer iguales pero al variar la parte del host, se convierten en dos usuarios diferentes. Por ejemplo, usuario@192.168.1.10 sería diferente de usuario@192.168.1.11

En MYSQL el nombre de usuario debe de cumplir:

- Tener un máximo de 16 caracteres
- Debe comenzar por letra
- No puede repetirse para el mismo host
- Si el usuario lleva espacios en blanco, se coloca entre comillas simples

Para la parte que se refiere al host, es posible usar:

- Direcciones IP, como ***192.168.1.10***
- Nombres de host, como ***localhost*** o cualquier otro nombre reconocido por nuestro servidor DNS
- Direcciones IP con máscara, como ***192.168.1.0/255.255.255.0*** (sólo son válidas de 8,16,24 o 32 bits)
- Direcciones IP con el comodín ***%,*** por ejemplo ***192.168.%.%*** representa cualquier máquina de la red 192.168.0.0
- En los dos últimos puntos anteriores, el host se escribe entre comillas simples.

Cada usuario tiene asociada una contraseña así como una serie de operaciones posibles para realizar.

(2.9.2) creación de usuarios

Desde la versión **5.0.2** de MySQL es posible utilizar el comando estándar **CREATE USER**. La sintaxis es:

```
CREATE USER usuario [IDENTIFIED BY [PASSWORD] 'contraseña'][, ...]
```

Se pueden crear usuarios sin indicar contraseñas. El nombre de usuario debe incluir el host (como se comentó en el apartado anterior), de otro modo usará el host '%' que representa a cualquier máquina (es un usuario global).

No es obligatorio el apartado IDENTIFIED BY que permite indicar la contraseña; si no se hace uso de él, la contraseña del usuario queda en blanco (situación nada recomendable). La contraseña se puede indicar en texto plano o a través de la función PASSWORD indicado el resultado de aplicar la función PASSWORD() (de esa forma se oculta el texto plano).

Los usuarios así creados no tienen privilegios asociados,

(2.9.3) borrado de usuarios

Se realiza mediante la instrucción:

```
DROP USER usuario [,...];
```

Si el usuario tiene sesión abierta, no se cierra la sesión. Se aplicará el comando al cierre de la sesión de dicho usuario.

(2.9.4) consulta de los usuarios de MySQL

La tabla mysql.user contiene la lista completa de usuarios. Modificar esta tabla permite modificar los usuarios, por lo que las instrucciones INSERT, DELETE o UPDATE en esta tabla añaden, modifican o eliminan usuarios; aunque no se recomienda ni INSERT ni DELETE (al existir las instrucciones de creación y eliminación de usuario estándares).

Las principales columnas de esa tabla son user, host y password.

(2.9.5) modificar usuarios de MySQL

El comando UPDATE sobre la tabla de usuarios, mysql.user, es la forma habitual de hacerlo, pero necesitamos invocar al comando FLUSH PRIVILEGES para que los cambios se realicen al instante. Ejemplos:

```
UPDATE mysql.user SET host='192.168.1.%' WHERE user='opersys';
UPDATE mysql.user SET password=PASSWORD('123456')
WHERE user='clara';
FLUSH PRIVILEGES;
```

(2.9.6) cambiar de nombre a un usuario

Se usa el comando no estándar, RENAME USER, de esta forma:

```
RENAME USER nombreAntiguo TO nombreNuevo
   [,nombreAntiguo2 TO nombreNuevo2 [,...]
```

(2.9.7) concesión de privilegios en MySQL

En MySQL es el comando estándar GRANT el que permite la concesión de privilegios. La sintaxis es extensa:

```
GRANT tipoDePrivilegio[(listaColumnas1)][, tipoDePrivilegio[(listaColumnas2)][,...]]
  ON [tipoDeObjeto]{tabla | * | *.* | baseDeDatos.* | baseDeDatos.tabla}
  TO usuario1 [IDENTIFIED BY [PASSWORD] 'contraseña'][, usuario2...]
  [WITH opción [,opción2[,...]]]
```

El *tipoDeObjeto* puede ser:

- TABLE
- FUNCTION
- PROCEDURE

Si no se indica tipo de objeto, se entiende que nos referimos a una tabla (que es lo habitual.

Las *opciones* del apartado WITH son:

- GRANT OPTION. Que permite que el usuario al que se le conceden los privilegios pueda, a su vez, concederles a otros.
- MAX_QUERIES_PER_HOUR *n*. Permite indicar el máximo número de consultas (*n*) a la hora que se le permiten al usuarios.
- MAX_UPDATES_PER_HOUR *n* Máximo número de operaciones de modificación de datos permitidas en una hora.
- MAX_CONNECTIONS_PER_HOUR *n*. Máximo número de conexiones que se le permiten hacer al usuario en una hora.
- MAX_USER_CONNECTIONS *n*. Conexiones concurrentes que como máximo el usuario puede mantener abiertas.

En todas las opciones anteriores si se le da a *n* un valor de *0*, entonces se traduce como **ilimitado**. Es decir MAX_USER_CONNECTIONS o permite tener ilimitadas conexiones simultáneas.

El objeto al que se le aplican los privilegios puede ser:

- **Una tabla de la base de datos actual**. De ella se indica el nombre simplemente.
- **Una tabla de una base de datos concreta**. Se indica la base de datos y la tabla separadas por un punto.
- **Todas las tablas de la base de datos actual**. Se indica un asterisco.
- **Todas las tablas de una base de datos**. Se indica el nombre de la base de datos seguida de un punto y un asterisco.
- **Todas las tablas de todas las bases de datos**. Dos asteriscos separados por un punto.

Además existen cinco niveles de aplicación de los permisos:

- **Nivel global**. Se aplican a todas las bases de datos. Por lo que se deben aplicar obligatoriamente al objeto ***.*** Ejemplos de privilegios que se aplican obligatoriamente a este nivel son **CREATE USER** o **SHUTDOWN**
- **Nivel de bases de datos**. Se aplican a todos los objetos de una base de datos. Se deben de aplicar usando el asterisco en las tablas pero un nombre de base de datos.
- **Nivel de tabla**. Se aplican sólo a una tabla.
- **Nivel de columna**. Se aplican a columnas de una tabla
- **Nivel de rutina**. Son privilegios que se aplican a rutinas.

La cláusula **IDENTIFIED BY** permite incluso crear el usuario al que se da permisos; de hecho antes de la versión 5 de MySQL era la forma de crear usuarios.

Los posibles permisos (apartado ***tiposDePrivilegio*** de la sintaxis de GRANT) que se pueden otorgar son:

Permiso	Significado
ALL [PRIVILEGES]	Otorga todos los privilegios
ALTER	Permite el uso de **ALTER TABLE**
ALTER ROUTINE	Modifica o borra rutinas almacenadas
CREATE	Permite crear tablas
CREATE ROUTINE	Crea rutinas almacenadas
CREATE TEMPORARY TABLES	Permite el uso de **CREATE TEMPORARY TABLE**
CREATE USER	Permite el uso de **CREATE USER, DROP USER, RENAME USER**, y **REVOKE ALL** PRIVILEGES
CREATE VIEW	Permite crear vistas
DELETE	Permite eliminar filas de tablas
DROP	DROP Permite el uso de **DROP TABLE**
EXECUTE	Permite ejecutar rutinas
FILE	Permite importar y exportar datos mediante **SELECT ... INTO OUTFILE** y **LOAD DATA INFILE**
INDEX	Permite crear y borrar índices
INSERT	Permite añadir filas
LOCK TABLES	Permite el uso de **LOCK TABLES** en tablas para las que tenga el permiso **SELECT**
PROCESS	Permite el uso de **SHOW FULL PROCESSLIST**
RELOAD	Permite el uso de **FLUSH**
REPLICATION CLIENT	Permite al usuario preguntar dónde están los servidores maestro o esclavo

Permiso	Significado
REPLICATION SLAVE	Necesario para los esclavos de replicación (para leer eventos del log binario desde el maestro)
SELECT	Permite consultar datos
SHOW DATABASES	Permite consular la lista completa de bases de datos
SHOW VIEW	Permite el uso de SHOW CREATE VIEW
SHUTDOWN	Permite el uso de mysqladmin shutdown, que cierra la instancia de MySQL
SUPER	Permite el uso de los comandos CHANGE MASTER, KILL, PURGE MASTER LOGS, y SET GLOBAL; además se permite al usuario que el comando mysqladmin debug le permita conectar (una vez) incluso si se ha superado el número máximo de conexiones.
UPDATE	Permite la modificación de datos
USAGE	Sin privilegios, el estado que tiene un usuario que se acaba de crear con CREATE USER.
GRANT OPTION	Permite dar permisos

Ejemplos:

```
GRANT INSERT,DELETE,UPDATE,SELECT ON almacen.pedidos TO ana;
#Permite a la usuaria ana@% permisos de modificación adición ,
#borrado y consulta sobre la tabla pedidos de la base de datos almacen
GRANT INSERT,DELETE,UPDATE,SELECT ON almacen.pedidos TO mario
WITH GRANT OPTION;
#Igual que la anterior para el usuario mario@% que además podrá el
#mismo conceder esos permisos
GRANT SELECT ON almacen.* TO felipe@192.168.1.32;
#al usuario indicado se le permite consultar todas las tablas del almacen
GRANT ALL ON almacen.* TO clara;
#clara puede hacer cualquier operación sobre la tabla de almacenes
GRANT CREATE ON almacen.* TO julian IDENTIFIED BY 'Caswq1209';
#Crea o modifica (si existe) el usuario Julian con la contraseña
indicada y permiso de  creación de tablas en la base de datos almacén
```

(2.9.8) revocación de permisos

El comando estándar REVOKE se encarga de quitar los permisos concedidos a un usuario concreto. La sintaxis del comando es:

```
REVOKE tipoDePrivilegio [(listaColumnas1)]
    [,tipoDePrivilegio[(listaDeColumnas2)]] [,...]
```

```
ON [tipoDeObjeto]{tabla | * | *.* | baseDeDatos.* | baseDeDatos.tabla}
FROM usuario1[, usuario2[,...]]
```

Ejemplo:

```
REVOKE SHUTDOWN ON *.* FROM alberto@localhost;
REVOKE INSERT,DELETE,UPDATE ON almacen.pedidos FROM clara;
REVOKE ALL PRIVILEGES, GRANT OPTION FROM ana, elena;
```

La última instrucción elimina todos los privilegios.

(2.9.9) mostrar información sobre usuarios y privilegios

La información sobre usuarios y privilegios se encuentra, fundamentalmente en las tablas **user** (lista de usuarios), **host** (lista de hosts) y **db** (bases de datos del sistema). La instrucción **DESCRIBE** seguida del nombre de la tabla (por ejemplo *mysql.db*) permite observar las columnas de la tabla y así saber qué consultar

en ellas.

Otras tablas interesantes son:

- **tables_priv**. Privilegios concedidos a las tablas, de cada tabla aparecen los usuarios que pueden operar con ella y los privilegios concretos que se les ha concedido.
- **columns_priv.** Privilegios concedidos a las columnas.

Todas las tablas anteriores se pueden manipular para conceder o quitar permisos, aunque no es muy lógico que esas operaciones se hagan así.