

TEMA 01
Ejercicio práctico 04
Privilegios de Administración, roles y mecanismos de autenticación

NOMBRE:

GRUPO:

FECHA DE ENTREGA:

CALIFICACION:

1.1. OBJETIVO

Comprender y poner en práctica los conceptos referentes a los privilegios de administración así como los diferentes mecanismos de autenticación que pueden emplearse en una base de datos.

1.2. VISTAS DEL DICCIONARIO DE DATOS ASOCIADAS CON LA BASE DE DATOS.

1.2.1. Revisión general del diccionario de datos.

Una de las partes más importantes de una base de datos es su diccionario. El diccionario está formado por un conjunto de tablas base y vistas que proporcionan metadatos para realizar la administración de la propia BD.

- Tablas base: Almacenan información acerca de la base de datos. Únicamente el propio manejador lee y escribe en estas tablas. El formato de estas tablas es poco amigable para ser accedidas por usuarios comunes.
- Vistas: Decodifican las complejidades de las tablas bases en un conjunto de vistas con información útil para los usuarios.

Típicamente las vistas del diccionario de datos están agrupadas en conjuntos. Cada uno de estos conjuntos a su vez está formado por 3 tipos de vistas que se distinguen por el prefijo: DBA_, USER_ y ALL_. Estos 3 grupos contienen prácticamente la misma información la cual es accesible dependiendo el tipo de usuario.

Prefijo	Acceso a Usuarios	Contenido
DBA_	Accesibles para usuarios con privilegios de administración	Todos los objetos.
ALL_	Todos los usuarios	Todos los objetos a los cuales el usuario tiene privilegios.
USER_	Todos los usuarios	Todos los objetos que le pertenecen al usuario.

No todos los conjuntos de vistas cuentan con estos 3 subgrupos.

- Existe una vista muy particular llamada **dictionary** la cual contiene la lista y una breve descripción de TODAS las vistas del diccionario de datos. La columna `table_name` contiene en nombre de la vista.
- Uno de sus usos principales es la búsqueda de vistas que pudieran contener cierta información de interés.

```
sys@jrcbd2> desc dictionary
Name          Null?     Type
-----
TABLE_NAME    VARCHAR2 (128)
COMMENTS      VARCHAR2 (4000)
```

1.2.2. Vistas dinámicas de desempeño del diccionario de datos.

Durante su operación, la base de datos mantiene a un conjunto de tablas virtuales que almacenan información de su actividad. Se les conoce como vistas dinámicas debido a que son continuamente actualizadas durante la operación de la propia BD. Este tipo de vistas inician con el prefijo V\$. La información que generalmente contiene este tipo de vistas son:

- Parámetros de sesión y del sistema
- Uso de memoria
- Estado o estatus de los archivos de la base de datos.
- Progreso de tareas programadas.
- Ejecución de sentencias SQL
- Estadísticas y métricas.

1.3. EJECUCIÓN DE SCRIPTS.

1.3.1. Usuario de ejecución

Todos los scripts SQL de los ejercicios prácticos del curso **NO** deben ser almacenados empleando al usuario `root` u `oracle` del sistema operativo. Los scripts deben crearse empleando el usuario administrador u ordinario con el que normalmente se inicia sesión gráfica en el equipo a excepción que el manual indique una instrucción particular o de excepción.

1.3.2. Carpeta de trabajo.

A nivel general durante el curso se generan 2 tipos de archivos: Scripts SQL que serán ejecutados en SQL*Plus y Shell scripts.

- Para el caso de archivos SQL generalmente son invocados por los siguientes usuarios
 - Por el usuario `oracle` del sistema operativo (se trata de evitar este usuario a medida de lo posible)
 - Por el usuario ordinario o administrador que fue creado durante el proceso de instalación (forma preferida).
- Para el caso de los Shell scripts, generalmente se invocan empleando los siguientes usuarios:
 - Por el usuario `root` del sistema operativo (se trata de evitar este usuario a medida de lo posible).
 - Por el usuario `oracle` del sistema operativo (se trata de evitar este usuario a medida de lo posible)
 - Por el usuario ordinario o administrador que fue creado durante el proceso de instalación (forma preferida).

Debido a que se trabaja con diversos usuarios, suele ser común caer en problemas de permisos tanto de lectura como de escritura, ya que por default cada uno de estos usuarios puede tener diferentes privilegios.

- Es posible crear un usuario y asignarle los privilegios necesarios para cubrir todos los escenarios anteriores. Por ejemplo, crear un usuario `admin`, asignarle privilegios de administración para hacer uso del comando `sudo`, y adicionalmente agregarlo al mismo grupo que el usuario `oracle` para modificar y acceder libremente a la base de datos. Esto se considera **mala práctica** y provoca graves problemas de seguridad. **No emplear** esta opción.
- Por default el usuario `oracle` NO cuenta con privilegios de administración del sistema operativo, no puede hacer uso del comando `sudo`. De forma similar, se considera **mala práctica** asignarle privilegios para hacer uso del comando `sudo`.

1.3.2.1. Ubicación de la carpeta de trabajo

Para solucionar estos inconvenientes se recomienda realizar el siguiente procedimiento:


1. Seleccionar un directorio de trabajo en el que se guardarán todos los scripts y en general, documentación del curso. A esta carpeta le llamaremos `curso_bda`. ¿En dónde será conveniente crearla? De preferencia, en una ubicación de tal forma que se cumplan con las siguientes condiciones:
 - La carpeta debería ubicarse en una partición diferente a las particiones donde se encuentran instalados todos los sistemas operativos del equipo. Esta opción protege a los archivos de una posible pérdida del sistema operativo o algún borrado accidental. En su defecto, se puede proteger con algún servicio en la nube (Google Drive, Dropbox, GitHub, etc.).
 - No seleccionar directorios que inicien con `/tmp`, `/home/<usuario>`. La primera ruta es una carpeta temporal cuyo contenido puede ser eliminado de forma periódica por el propio sistema operativo. La segunda carpeta no permite el acceso a usuarios diferentes al usuario propietario, se considera **mala práctica** cambiar los permisos a las carpetas `home` de cada usuario.

Para efectos prácticos, seleccionar alguno de los siguientes escenarios.

Escenario 1: Instalación nativa con una partición NTFS existente que puede ser compartida tanto en Linux como en Windows. En caso de no existir, se puede crear empleando GParted.

A. Identificar a la partición NTFS empleando el comando `lsblk`

- En la siguiente imagen se muestra la salida del comando. En el equipo se cuentan con 2 discos `sda` y `sdb`. Se observan varias particiones `sda*` y `sdb*`. Suponer que se ha decidido guardar los documentos de la materia en la partición `sda8`.
- Por default este tipo de particiones no están disponibles para realizar operaciones de lectura y escritura, requieren ser cargadas en algún punto de montaje. La opción más recomendable es agregar una línea en el archivo `/etc/fstab` para que la partición esté disponible cuando se inicia el sistema.





```
[jorge@lap-red-ora ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda         8:0    0 119.2G 0 disk
├─sda1      8:1    0  100M 0 part /boot/efi
├─sda2      8:2    0   16M 0 part
├─sda3      8:3    0  60.4G 0 part /media/jorge/OS-WIN
├─sda4      8:4    0   1.8G 0 part
├─sda5      8:5    0   32G 0 part /media/jorge/OS-MINT
├─sda6      8:6    0    5G 0 part [SWAP]
├─sda7      8:7    0   20G 0 part /
└─sdb       8:16   0 931.5G 0 disk
   ├─sdb1    8:17   0  100G 0 part /media/jorge/PROYS
   ├─sdb2    8:18   0  100G 0 part /media/jorge/DOCS
   ├─sdb3    8:19   0  100G 0 part /media/jorge/FAM
   ├─sdb4    8:20   0  100G 0 part /media/jorge/MUSIC
   ├─sdb5    8:21   0  100G 0 part /media/jorge/SOFT
   ├─sdb6    8:22   0   40G 0 part /media/jorge/DOWN
   ├─sdb7    8:23   0  150G 0 part /media/jorge/VIRTUAL
   ├─sdb8    8:24   0   40G 0 part
   ├─sdb9    8:25   0   55G 0 part /media/jorge/U01-MINT
   ├─sdb10   8:26   0  56.5G 0 part /u01
   ├─sdb11   8:27   0   40G 0 part /media/jorge/OS-FREE
   └─sdb12   8:28   0   50G 0 part /media/jorge/BACKUP-LX
```

- Se recomienda agregar una línea similar a la siguiente

```
# Particion empleada para BDA
/dev/sdb8 /media/jorge/BACKUP ntfs-3g nouser,exec,rw,locale=es_MX.utf8,umask=077,gid=1000,uid=1000 0 0
```

La línea está formada por 6 configuraciones separadas por espacios o tabulaciones.

- Los únicos valores que se deben modificar son las rutas de las 2 primeras configuraciones según corresponda.
- La primera sección indica la partición o dispositivo
- La segunda sección indica el punto de montaje donde se podrá acceder a su contenido. Típicamente se emplea la carpeta `/media/<usuario>/<nombre_particion>`. En este ejemplo se ha decidido nombrar a la partición `BACKUP`; se puede emplear cualquier nombre.
- La tercera sección indica el tipo de partición.
- La cuarta sección está formada por una serie de parámetros. Entre los más destacados está `umask=077`, indica que todas las carpetas y archivos en esta partición tendrían los permisos 755 (notar que la notación de `umask` es diferente, `077` equivale a `755`). Esta configuración permite todo el control para el dueño así como permisos de lectura y ejecución para todos los demás. Dicha configuración permite resolver todos los problemas de permisos, útil para los propósitos del curso.
- En la cuarta columna observar `gid=1000,uid=1000`; Estos 2 valores indican el identificador del grupo y del usuario dueño de todos los archivos de esta partición. Este usuario debe corresponder con el usuario ordinario configurado durante el proceso de instalación. Típicamente al ser el primer usuario se le asigna el id 1000. Para verificar este valor abrir una terminal con el usuario ordinario y ejecutar el comando `id`, comprobar que los ids asignados sean los correctos:

```
[jorge@lap-red-ora unam-clases]$ id
uid=1000(jorge) gid=1000(jorge) groups=1000(jorge),10(wheel)
```

- Antes de probar esta configuración se deberá crear la estructura de carpetas que representa al punto de montaje. Ejemplo:

```
sudo mkdir -p /media/jorge/BACKUP
```

- Finalmente, ejecutar la siguiente instrucción (o en su defecto reiniciar) para que los cambios tomen efecto:

```
sudo mount -a
```

- Para verificar, ejecutar el comando `df -h | grep BACKUP` (Cambiar el nombre según corresponda), se deberá obtener una salida similar a la siguiente:

```
/dev/sdb8          40G   21G   20G   52% /media/jorge/BACKUP
```

La salida anterior indica que el dispositivo fue montado en la ruta configurada y tiene un 52% de espacio ocupado.

- Crear la carpeta `curso_bda`

```
mkdir /media/jorge/BACKUP/curso_bda
```

Tip: Cabe mencionar que las particiones NTFS tienen la siguiente restricción: Una vez que se carga el dispositivo en el punto de montaje, el dueño, grupo y permisos no pueden ser modificados. Para efectos del curso esto **no representa** problema alguno.

Escenario 2: Instalación nativa con una partición compartida con formato ext4.

El procedimiento es similar. La única diferencia es la línea agregada en `/etc/fstab`. Suponer que `sdb8` tiene formato ext4:

```
# BACKUP was on /dev/sdb8 during installation
/dev/sdb8 /media/jorge/BACKUP ext4    nouser,exec 0      0
```

- Como se puede observar, su configuración es más simple.

Escenario 3: Instalación con máquina virtual. Uso de una carpeta compartida configurada en VirtualBox. Las carpetas compartidas a través de VirtualBox empleando el procedimiento descrito en ejercicios anteriores permite tener acceso a una carpeta de la máquina anfitriona desde Linux. Si se desea almacenar todos los documentos del curso, realizar las siguientes configuraciones:

1. Configurar una carpeta compartida en caso de no existir tal cual se describe en el primer ejercicio práctico.
2. Por default todas las carpetas y archivos de esta carpeta compartida le pertenecen al usuario `root` y al grupo `vboxsf`. Debido a que el usuario ordinario creado durante el proceso de instalación se le asignó de forma adicional el grupo `vboxsf`, dicho usuario no tendrá problemas de permiso. Sin embargo, el usuario `oracle` no tiene acceso. Para corregir este inconveniente ejecutar la siguiente instrucción para agregar al usuario `oracle` al grupo `vboxfs`.

```
sudo usermod -G vboxsf,dba,oper oracle
```

Para comprobar, los grupos asignados mostrar la salida del comando `id` para el usuario `oracle` y el usuario ordinario según corresponda.

```
id oracle
uid=54321(oracle) gid=54321(oinstall) groups=54321(oinstall),983(vboxsf),54322(dba),54323(oper)
```

```
id jorge
uid=1000(jorge) gid=1000(jorge) groups=1000(jorge),10(wheel),983(vboxsf)
```

Notar que ambos usuarios son miembros del grupo `vboxsf`.

3. Crear la carpeta `curso_bda` dentro de la carpeta compartida empleando el usuario ordinario.

Escenario 4. Crear una carpeta cualquiera en Linux (no compartida con otros sistemas operativos). Representa la forma más fácil:

```
sudo mkdir -p /media/jorge/curso_bda
sudo chmod -R 775 /media/jorge/curso_bda
```

1.3.2.2. Estructura interna de la carpeta de trabajo.

- Por último, se recomienda crear una carpeta por cada tema y ejercicio práctico dentro de `curso_bda`.

```
mkdir tema01
mkdir tema01/e-practico-01
..
..
```

1.4. VERSIÓN DE LA BASE DE DATOS.

A partir de este ejercicio práctico se hará un uso extensivo de las vistas del diccionario de datos de la BD. La forma más sencilla para familiarizarse con estas vistas es empleando la documentación oficial que incluye su estructura, descripción y razón de ser. La documentación puede ser consultada en la siguiente liga (Ajustar la versión en caso de ser necesario) <https://docs.oracle.com/en/database/oracle/oracle-database/19/refrn/index.html>

Crear un script llamado `s-01-version-bd.sql` que realice las siguientes acciones.

- Conectar como `sysdba`
- Crear un usuario llamado `<nombre>0104` en caso de no existir. Asignarle privilegios para crear sesiones y tablas, cuota ilimitada en el tablespace `users`.

Una forma sencilla para implementar estos 2 primeros puntos es crear un pequeño código PL/SQL que verifique la existencia del usuario. Si el usuario existe, el script lo va a eliminar de tal forma que al invocar la creación el usuario el programa no falle por la existencia previa del usuario. Este programa

puede emplearse de forma similar en todos los ejercicios prácticos. Al eliminar al usuario, permite que el script pueda ser ejecutado N veces sin generar errores de objetos existentes.

```
--esta instrucción permite detener la ejecución del script al primer error
--útil para detectar errores de forma rápida.
--al final del script se debe invocar a whenever sqlerror continue none
--para regresar a la configuración original.
whenever sqlerror exit rollback;

-- para propósitos de pruebas y propósitos académicos se incluye el password
-- no hacer esto en sistemas reales.
Prompt conectando como usuario sys
connect sys/system1 as sysdba

declare
  v_count number;
  v_username varchar2(20) := 'JORGE0104';
begin
  select count(*) into v_count from all_users where username=v_username;
  if v_count >0 then
    execute immediate 'drop user '||v_username|| 'cascade';
  end if;
end;
/

Prompt creando al usuario JORGE0104
create user jorge0104 identified by jorge quota unlimited on users;
grant create session, create table to jorge0104;
```

- Realizar una consulta al diccionario de datos considerando la vista `product_component_version`.
- Crear una tabla que contenga los datos obtenidos de la consulta anterior. Tip: En SQL es posible crear una tabla a partir del resultado de una consulta.
- Por default, las vistas que contienen datos del funcionamiento de la instancia o de su configuración solo pueden ser accedidas por el usuario SYS. Por esta razón el script debe ser invocado por este usuario. Para hacer que la tabla le pertenezca al usuario `<nombre>0104` Se debe incluir el nombre del esquema en la definición de la tabla. De esta forma la tabla será creada y poblada con los datos de la consulta y asignada al usuario `<nombre>0104` El nombre de la tabla y columnas a seleccionar se muestra a continuación:

```
create table <nombre>0104.t01 db version as
select product,version,version_full
from product_component_version;
```

- Consultar los datos de la tabla para verificar resultados.
- No olvidar agregar el encabezado al script:

```
--@Autor:          <Nombre del alumno>
--@Fecha creación: dd/mm/yyyy
--@Descripción:    <Descripción corta del script>
```

1.5. EXPLORANDO EL ROL DBA

1.5.1. Lista de roles existentes

Crear un script SQL llamado `s-02-roles.sql` que realice las siguientes acciones.

- Considerar la vista del diccionario `dba_roles` la cual contiene la lista de todos los roles que se crean por defecto cuando se instala la base de datos, obtener el identificador del rol y su nombre.
- A partir de esta consulta crear una tabla que contenga las 2 columnas anteriores llamada `<nombre>0104.t02_db_roles`

1.5.2. Lista de privilegios asignados al rol DBA.

Considerando la vista `dba_sys_privs` la cual contiene la lista de privilegios que se le han otorgado al rol DBA generar una tabla `<nombre>0104.t03_dba_privs` que contenga una columna llamada `privilege`. La tabla solo debe incluir a los privilegios que se le asignaron al rol DBA.

1.6. ARCHIVO DE PASSWORDS.

Generar un Shell script llamado `s-03-archivo-passwords.sh`. El archivo deberá ser invocado por el usuario Oracle y deberá realizar las siguientes acciones:

- En caso de no existir, realizar el respaldo del archivo de passwords en `/home/${USER}/backups`
- Una vez que el respaldo ha sido creado, el script deberá eliminar el archivo de passwords de la BD para simular su pérdida.
- Empleando el comando correspondiente, generar un nuevo archivo de passwords. Incluir a los usuarios `SYS` y `SYSBACKUP`. Configurar el comando para que el archivo sobrescriba en caso de existir otro archivo creado previamente.
- Asignar el password `Hola1234#` para ambos usuarios.

1.7. USUARIOS DE ADMINISTRACIÓN

Crear un script `s-04-privs-admin.sql` el cual será invocado por el usuario `sys` empleando el password asignado en ejercicio anterior: `Hola1234#`. El script deberá realizar las siguientes tareas:

- En caso de no existir, crear a los usuarios `<nombre>0105` y `<nombre>0106` sin cuota de almacenamiento. Asignar como password el valor `<nombre>`.
- Realizar las configuraciones necesarias para que ambos usuarios puedan crear sesiones (no se requiere otro privilegio).
- Asignar los siguientes roles de administración:
 - `sysdba` al usuario `<nombre>0104`
 - `sysoper` al usuario `<nombre>0105`
 - `sysbackup` al usuario `<nombre>0106`

1.8. ESQUEMAS DE LOS PRIVILEGIOS DE ADMINISTRACIÓN.

Crear un script llamado `s-05-schemas.sql`. El script deberá ser invocado por el usuario `sys`. Deberá realizar las siguientes acciones:

- Entrar a sesión con el usuario `<nombre>0104` sin hacer uso de su privilegio de administración. Crear una tabla con la siguiente estructura:

```
create table t04_my_schema (
  username varchar2(128)
  schema_name varchar2(128)
)
```

- Otorgar los privilegios necesarios para que los 3 usuarios `<nombre>0104`, `<nombre>0105`, `<nombre>0106` puedan insertar registros en esta tabla. La inserción se realizará **únicamente** cuando estos 3 usuarios hayan iniciado sesión con su privilegio de administración. Esto significa que el privilegio a otorgar deberá considerar al usuario que se adquiere durante la sesión iniciada con dicho privilegio de administración.

Realizar las siguientes acciones para cada uno de los 3 usuarios

- Entrar a sesión con su correspondiente privilegio de administración.
- Realizar la inserción en la tabla `t04_my_schema` indicando el nombre del usuario y el nombre del esquema del usuario en turno.
- Hacer `commit`.

Ejemplo:

```
Prompt insertando con jorge0104 as sysdba
connect jorge0104/jorge as sysdba
insert into jorge0104.t04_my_schema values (
  sys_context('USERENV','CURRENT_USER'),
  sys_context('USERENV','CURRENT_SCHEMA')
);
--Realizar lo mismo para los otros 2 usuarios.
```

Posteriormente, el script deberá hacer una consulta a la vista `v$pwfile_users` para mostrar los siguientes datos de los usuarios que cuentan con privilegios de administración: nombre del usuario, banderas que indican los privilegios `sysdb`, `sysoper` y `sysbackup`; fecha del último login realizado. Los datos de esta consulta deberán ser mostrados por el script.

Finalmente, actualizar el password del usuario `sys` al valor que se tenía previamente configurado: `system1`.

1.9. VALIDADOR.

- Obtener todos los archivos de la carpeta correspondiente a este ejercicio práctico. Copiarlos a la misma carpeta donde se encuentra el programa.
- Ejecutar el validador:

```
su -l oracle
```

```
export ORACLE_SID=jrcbda1
sqlplus /nolog
start s-06-validador-oracle-main.sql
```

1.10. CONTENIDO DE LA ENTREGA.

No es necesario imprimir o entregar todas las instrucciones incluidas en este documento. Entregar solo los siguientes puntos:

- C1. Código del programa script s-02-roles.sql
- C2. Código del programa script s-03-archivo-passwords.sh
- C3. Código del programa script s-05-schemas.sql
- C4. Salida de ejecución del validador.
- Elementos generales indicados en la rúbrica general de ejercicios prácticos (datos generales, conclusiones y comentarios).
- Entrega individual