Using Privilege Analysis ( capture privilege )

## Create Users, Roles, Tables and Grant Privileges and Roles

Use SQL\*Plus to connect as system to create the users JIM and TOM , roles, and grant privileges and or roles.

Connect as SYSTEM:

**. oraenv**

**[enter cdb1 at the prompt]**

**sqlplus system/oracle@localhost:1521/pdb1**

Create users.

**create user jim identified by jim;  
create user tom identified by tom;  
create user ann identified by ann;**

Create roles.

**create role HR\_MGR;  
create role SALES\_CLERK;   
create role ANALYST;**

Grant privileges.

**grant create session to jim, tom, ann;**

**grant select, update, delete, insert on hr.employees to HR\_MGR;  
grant HR\_MGR to JIM;  
  
grant select on sh.sales to SALES\_CLERK;  
grant SALES\_CLERK to TOM;   
  
grant select any table to ANALYST;  
grant ANALYST to ann;**

## Define the Captures

In this section, you will prepare the three types of captures.

### Define the capture of privileges used by all users.

Define the capture of privileges used by all users. Use the DBMS\_PRIVILEGE\_CAPTURE package and the CREATE\_CAPTURE procedure with the appropriate type of capture.

**exec SYS.DBMS\_PRIVILEGE\_CAPTURE.CREATE\_CAPTURE ( -   
        name           =>  'All\_privs', -   
        description    =>  'All privs used', -  
        type           =>   dbms\_privilege\_capture.g\_database)**

### Define the capture of privileges used through roles.

Define a capture of privileges used by roles HR\_MGR, ANALYST, and SALES\_CLERK. Use the DBMS\_PRIVILEGE\_CAPTURE package and the CREATE\_CAPTURE procedure with the appropriate type of capture and the list of roles analyzed.

**exec SYS.DBMS\_PRIVILEGE\_CAPTURE.CREATE\_CAPTURE ( -   
        name           => 'Role\_privs', -  
        description    => 'Privs used by HR\_MGR, SALES\_CLERK', -  
        type           =>  dbms\_privilege\_capture.g\_role, -  
        roles          =>  role\_name\_list('HR\_MGR', 'SALES\_CLERK','ANALYST'))**

### Define the capture of privileges used through contexts.

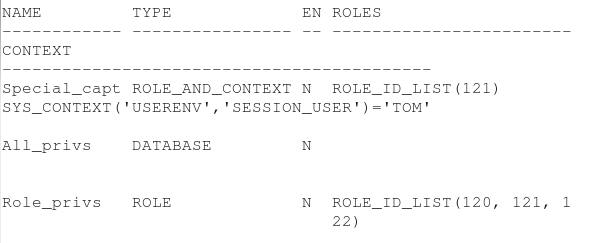
Define a capture of privileges used by the user TOM or by the specific role SALES\_CLERK.

Use the DBMS\_PRIVILEGE\_CAPTURE package and the CREATE\_CAPTURE procedure with the appropriate type of capture, the list of roles analyzed and the context in which the analysis would take place once started.

**exec SYS.DBMS\_PRIVILEGE\_CAPTURE.CREATE\_CAPTURE ( -   
        name           => 'Special\_capt', -   
        description    => 'Special', -  
        type           =>  dbms\_privilege\_capture.g\_role\_and\_context, -  
        roles          =>  role\_name\_list('SALES\_CLERK'), -  
        condition      => 'SYS\_CONTEXT(''USERENV'',''SESSION\_USER'')=''TOM''')**

### List the existing captures.

Use the DBA\_PRIV\_CAPTURES view.  
**Note:** You may see different role ID results.   
**COL name     FORMAT A12  
COL type     FORMAT A16  
COL enabled  FORMAT A2  
COL roles    FORMAT A24** **COL context  FORMAT A43  
  
select name, type, enabled, roles, context  
from   dba\_priv\_captures;**



## Start Privilege Captures and Analyze

### Start and analyze the capture of privileges used by all users.

Start capturing the privileges while users are performing their daily work using privileges. Use the ENABLE\_CAPTURE procedure.  **exec SYS.DBMS\_PRIVILEGE\_CAPTURE.ENABLE\_CAPTURE (name => 'All\_privs')  
  
exit  
[Note that you must exit SQL\*Plus and then relaunch SQL\*Plus in the next step, otherwise you may see unintended results in the privilege analysis views in later steps of this tutorial.]**

The users JIM, TOM, and ANN run SQL statements using privileges. JIM, and ANN who select rows from HR.EMPLOYEES table and TOM who selects rows from SH.SALES table.

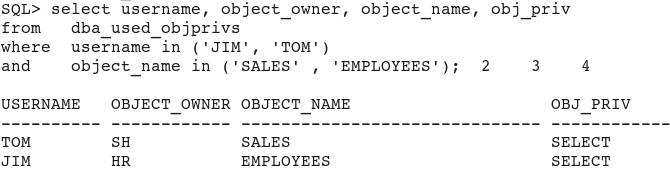
**sqlplus jim/jim@localhost:1521/pdb1  
select \* from hr.employees where salary < 3000;  
  
connect tom/tom@localhost:1521/pdb1  
select \* from sh.sales where amount\_sold < 6.42 and cust\_id = 6452;  
  
connect ann/ann@localhost:1521/pdb1  
select \* from hr.employees where salary > 3000 and rownum <=5 order by salary desc;**

**exit  
[Note you must exit SQL\*Plus and then relaunch SQL\*Plus in the next step.]**

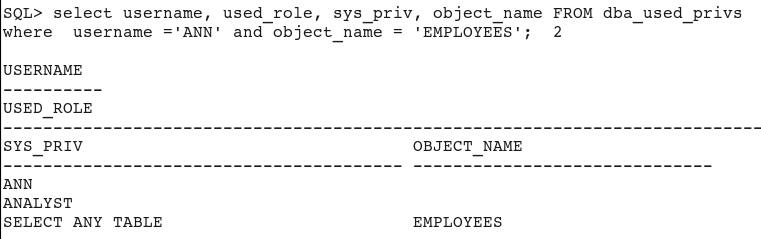
Stop capturing. Use the DISABLE\_CAPTURE procedure. 

**sqlplus system/oracle@localhost:1521/pdb1  
exec SYS.DBMS\_PRIVILEGE\_CAPTURE.DISABLE\_CAPTURE (name => 'All\_privs')**

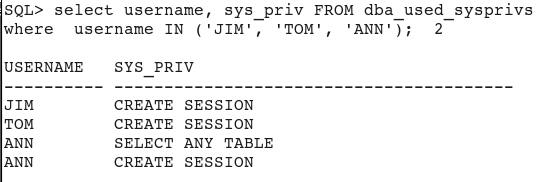
Generate the capture results. It may take a few minutes. Use the GENERATE\_RESULT procedure.  **exec SYS.DBMS\_PRIVILEGE\_CAPTURE.GENERATE\_RESULT (name => 'All\_privs')**Display the object privileges used during the capture period. Use the DBA\_USED\_OBJPRIVS view.  **COL username FORMAT A10  
COL object\_owner FORMAT A12  
COL object\_name FORMAT A30  
COL obj\_priv FORMAT A25  
  
select username, object\_owner, object\_name, obj\_priv  
from   dba\_used\_objprivs  
where  username in ('JIM', 'TOM')  
and    object\_name in ('SALES' , 'EMPLOYEES');**



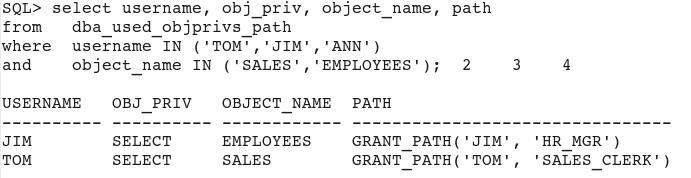
Display the privileges used by ANN. Use the DBA\_USED\_PRIVS view.  Does Ann, or the ANALYST role, really need the SELECT ANY TABLE privilege? **select username, used\_role, sys\_priv, object\_name FROM dba\_used\_privs  
where  username ='ANN' and object\_name = 'EMPLOYEES';**



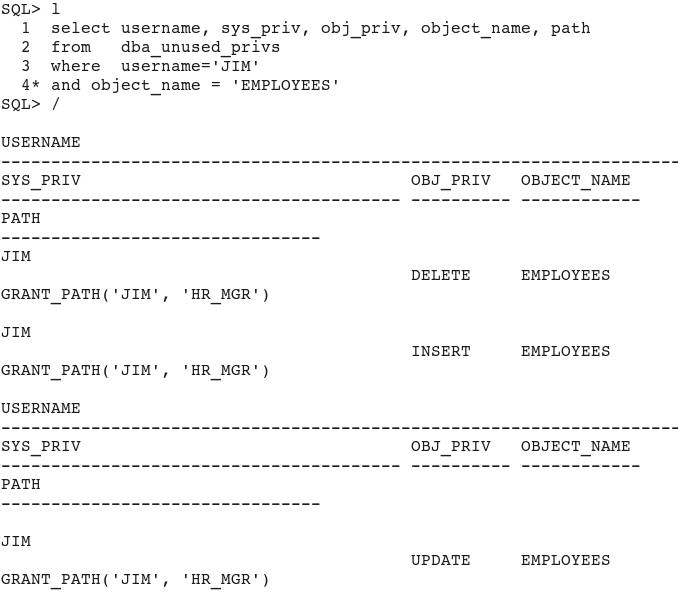
Display the system privileges used. Use the DBA\_USED\_SYSPRIVS view.  **select username, sys\_priv FROM dba\_used\_sysprivs  
where  username IN ('JIM', 'TOM', 'ANN');**



Display the path of the privileges used if the privileges were granted to roles, and roles to users. Use the DBA\_USED\_OBJPRIVS\_PATH view.  **COL object\_name FORMAT A12  
COL path FORMAT A32  
COL obj\_priv FORMAT A10  
  
select username, obj\_priv, object\_name, path  
from   dba\_used\_objprivs\_path   
where  username IN ('TOM','JIM','ANN')   
and    object\_name IN ('SALES','EMPLOYEES');**



JIM is granted select, update, delete, insert privileges on HR.EMPLOYEES table through HR\_MGR role. He used only the SELECT privilege thus far.  
Use the DBA\_UNUSED\_PRIVS view to list the unused privileges. You can decide which of the unused privileges can be revoked if necessary.  **select username, sys\_priv, obj\_priv, object\_name, path  
from   dba\_unused\_privs  
where  username='JIM'  
and object\_name = 'EMPLOYEES';**



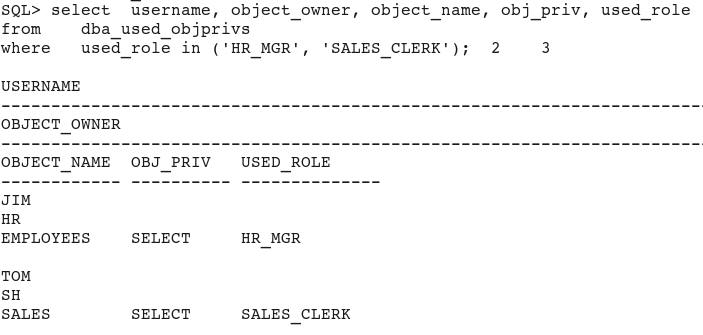
### Start and analyze the capture of privileges used through roles.

Delete the previous capture so as to remove all previous captured information from the views. Use the DROP\_CAPTURE procedure.  **exec SYS.DBMS\_PRIVILEGE\_CAPTURE.DROP\_CAPTURE (name => 'All\_privs')**Verify that there is no data left from the All\_privs capture. Use the DBA\_UNUSED\_PRIVS view. There should be no rows returned by the query. **select username, sys\_priv, obj\_priv, object\_name, path  
from   dba\_unused\_privs  
where  username='JIM';**Start capturing the privileges while users are performing their daily work using roles.  **exec SYS.DBMS\_PRIVILEGE\_CAPTURE.ENABLE\_CAPTURE (name => 'Role\_privs')**

The users JIM and TOM run SQL statements using privileges. JIM who selects rows from HR.EMPLOYEES table and TOM who selects rows from SH.SALES table.

**connect jim/jim@localhost:1521/pdb1  
select \* from hr.employees where salary < 3000;  
  
connect tom/tom@localhost:1521/pdb1  
select \* from sh.sales where amount\_sold < 6.42 and cust\_id = 6452;**

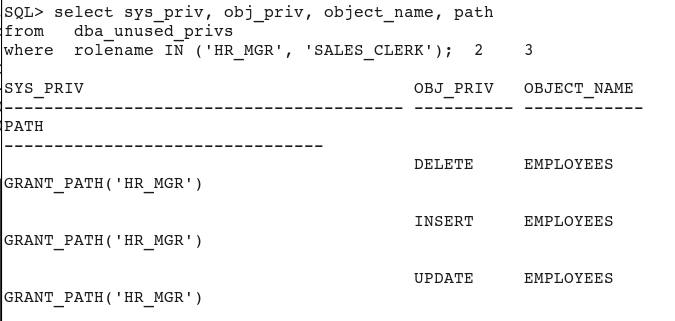
Stop capturing.   
  
**connect system/oracle@localhost:1521/pdb1  
exec SYS.DBMS\_PRIVILEGE\_CAPTURE.DISABLE\_CAPTURE (name => 'Role\_privs')**Generate the capture results. **exec SYS.DBMS\_PRIVILEGE\_CAPTURE.GENERATE\_RESULT (name =>'Role\_privs')**Display the object privileges used by the roles HR\_MGR and SALES\_CLERK during the capture period. **COL used\_role FORMAT A14**  
  
**select  username, object\_owner, object\_name, obj\_priv, used\_role  
from    dba\_used\_objprivs  
where   used\_role in ('HR\_MGR', 'SALES\_CLERK');**



Display the system privileges used by the roles HR\_MGR and SALES\_CLERK during the capture period.  **select username, sys\_priv, used\_role   
from dba\_used\_sysprivs  
where used\_role in ('HR\_MGR', 'SALES\_CLERK');**

No rows are returned because no system privileges were used.

HR\_MGR is granted select, update, delete on HR.EMPLOYEES table. The role used by JIM during the capture period used the SELECT privilege.  
The unused privileges are visible in DBA\_UNUSED\_PRIVS view. You can decide which of the unused privileges or role can be revoked if necessary.  **COL username FORMAT A12  
COL path FORMAT A32  
COL object FORMAT A10  
COL sys\_priv FORMAT A10  
COL obj\_priv FORMAT A10  
  
select sys\_priv, obj\_priv, object\_name, path  
from   dba\_unused\_privs  
where  rolename IN ('HR\_MGR', 'SALES\_CLERK');**



### Define the capture of privileges used through contexts.

Delete the previous capture so as to remove all captured information from the views.

**exec SYS.DBMS\_PRIVILEGE\_CAPTURE.DROP\_CAPTURE (name => 'Role\_privs')**

Verify that there is no data left from the Role\_privs capture. There should be no rows returned by the query. **select sys\_priv, obj\_priv, object\_name, path  
from   dba\_unused\_privs  
where  rolename IN ('HR\_MGR', 'SALES\_CLERK');**

Start capturing the privileges while users are performing their daily work using privileges.

**exec SYS.DBMS\_PRIVILEGE\_CAPTURE.ENABLE\_CAPTURE (name => 'Special\_capt')**

The users JIM and TOM run SQL statements using privileges. JIM selects rows from HR.EMPLOYEES table and TOM selects rows from SH.SALES table.

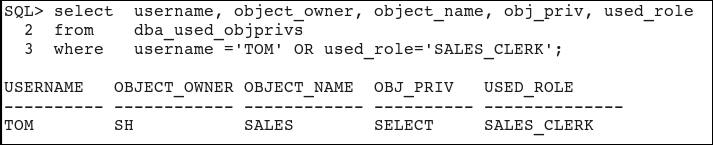
**connect jim/jim@localhost:1521/pdb1  
select \* from hr.employees where salary < 3000;  
  
connect tom/tom@localhost:1521/pdb1  
select \* from sh.sales where amount\_sold < 6.42 and cust\_id = 6452;**

Stop capturing.

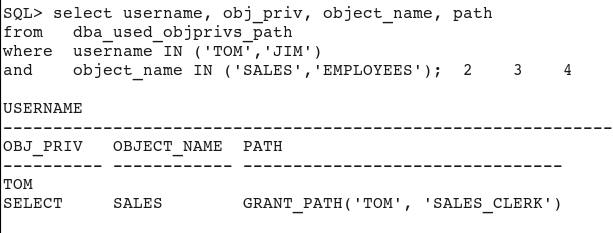
**connect system/oracle@localhost:1521/pdb1  
exec SYS.DBMS\_PRIVILEGE\_CAPTURE.DISABLE\_CAPTURE (name =>'Special\_capt')**

Generate the capture results.

**exec SYS.DBMS\_PRIVILEGE\_CAPTURE.GENERATE\_RESULT (name => 'Special\_capt')**Display the object privileges used during the capture period.  **COL username FORMAT A10  
COL owner FORMAT A8  
COL object FORMAT A16  
COL obj\_priv FORMAT A10  
COL used\_role FORMAT A14  
  
select  username, object\_owner, object\_name, obj\_priv, used\_role  
from    dba\_used\_objprivs  
where   username ='TOM' OR used\_role='SALES\_CLERK';**



Check whether any system privileges were used. There should be no rows returned, because no system privileges were used.  **select username, sys\_priv from dba\_used\_sysprivs;**Display the path of the privileges used if the privileges were granted to roles, and roles to users. Use the DBA\_USED\_OBJPRIVS\_PATH view.  **COL object FORMAT A12  
COL path FORMAT A32  
COL obj\_priv FORMAT A10  
  
select username, obj\_priv, object\_name, path  
from   dba\_used\_objprivs\_path   
where  username IN ('TOM','JIM')   
and    object\_name IN ('SALES','EMPLOYEES');**



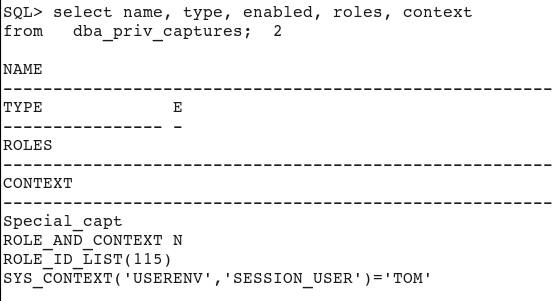
TOM is granted the select privilege on the SH.SALES table through SALES\_CLERK role. He used the privilege.  
The unused privs are visible in DBA\_UNUSED\_PRIVS view.  
The query returns no rows, because there are no unused privileges. Therefore, there is no privilege that has been unnecessarily granted.

**select username, sys\_priv, obj\_priv, object\_name, path  
from   dba\_unused\_privs  
where  username='TOM' OR rolename='SALES\_CLERK';**

## Delete Captures

List all captures to delete:

**COL name     FORMAT A12  
COL type     FORMAT A12  
COL enabled  FORMAT A2  
COL roles    FORMAT A26  
  
select name, type, enabled, roles, context  
from   dba\_priv\_captures;**



Delete all captures.

**exec SYS.DBMS\_PRIVILEGE\_CAPTURE.DROP\_CAPTURE (name => 'Special\_capt')**

## Resetting Your Environment

Delete users and roles.

**drop user jim cascade;  
drop user tom cascade;  
drop user ann cascade;  
  
drop role hr\_mgr;  
drop role sales\_clerk;  
drop role analyst;**