consultant.mubashir@gmail.com

Checking Users Status

SQL> DESC DBA_USERS;

SQL> SELECT USERNAME, PROFILE, ACCOUNT_STATUS FROM DBA_USERS;

DBA_USERS describes all users of the database, and contains more columns than ALL_USERS.

USER_USERS describes the current user, and contains more columns than ALL_USERS.

Checking Privileges

Querying DBA/USER Privilege Views

A database administrator (DBA) for Oracle can simply execute a query to view the rows in DBA_SYS_PRIVS, DBA_TAB_PRIVS, and DBA_ROLE_PRIVS to retrieve information about user privileges related to the system, tables, and roles, respectively.

For example, a DBA wishing to view all system privileges granted to all users would issue the following query:

SQL> SELECT * FROM DBA_SYS_PRIVS;

The DBA_SYS_PRIVS view contains three columns of data:

GRANTEE is the name, role, or user that was assigned the privilege.

PRIVILEGE is the privilege that is assigned.

ADMIN_OPTION indicates if the granted privilege also includes the ADMIN option.

To determine which users have direct grant access to a table we will use the DBA_TAB_PRIVS view:

SQL>SELECT * FROM DBA_TAB_PRIVS;

You can check the official documentation for more information about the columns returned from this query, but the critical columns are:

GRANTEE is the name of the user with granted access.

TABLE_NAME is the name of the object (table, index, sequence, etc).

PRIVILEGE is the privilege assigned to the GRANTEE for the associated object.

Finally, querying the DBA_ROLE_PRIVS view has much of the same information but applicable to roles instead, where the GRANTED_ROLE column specifies the role in question:

SQL> SELECT * FROM DBA_ROLE_PRIVS;

QUERYING THE CURRENT USER'S PRIVILEGES

If DBA access is not possible or necessary, it is also possible to slightly modify the above queries to view the privileges solely for the current user.

This is done by alternatively querying USER_ versions of the above DBA_ views. Thus, instead of looking at DBA_SYS_PRIVS we would query USER_SYS_PRIVS, like so:

SQL > SELECT * FROM USER_SYS_PRIVS;

Since the USER_ privilege views are effectively the same as their DBA_ counterparts, but specific to the current user only, the type of returned data and column names are all identical to those when querying DBA_ views instead