About User Accounts

The areas in which you can configure security are as follows:

• User accounts

For users to access your database, you must create user accounts and grant appropriate database access privileges to those accounts.
A user account is identified by a user name and defines the attributes of the user, including the following:

- Authentication method
- Password for database authentication
- Default tablespaces for permanent and temporary data storage
- Tablespace quotas
- Account status (locked or unlocked)
- Password status (expired or not)

When you create a user account, you must not only assign a user name, a password, and default tablespaces for the account, but you must also do the following:

- Grant the appropriate system privileges, object privileges, and roles to the account.
- If the user will be creating database objects, then give the user account a space usage quota on each tablespace in which the objects will be created.

Predefined User Accounts

- User accounts that are automatically created upon installation.
- All databases include the administrative accounts SYS, SYSTEM, SYSMAN, and DBSNMP.
- Administrative accounts are highly privileged accounts, and are needed only by individuals authorized to perform administrative tasks such as starting and stopping the database, managing database memory and storage, creating and managing database users, and so on.
- You log in to Oracle Enterprise Manager Database Control (Database Control) with SYS, SYSTEM, or SYSMAN. The Management Agent of Database Control uses the DBSNMP account to monitor and manage the database. You assign the passwords for these accounts when you create the database with Oracle Database Configuration Assistant (DBCA). You must not delete these accounts.
- Your database may also include **sample schemas**, which are a set of interlinked schemas that enable Oracle documentation and Oracle instructional materials to illustrate common database tasks. These schemas also provide a way for you to experiment without endangering production data.

Who Can Create User Accounts?

Users who has been granted the CREATE USER system privilege can create user accounts.
If you want to create users who themselves have the privilege to create users, then include the WITH ADMIN OPTION clause in the GRANT statement.

For example:

GRANT CREATE USER TO lbrown WITH ADMIN OPTION;

Creating a New User Account That Has Minimum Database Privileges

When you create a new user account, you should enable this user to access the database.

1. Use the CREATE USER statement to create a new user account.

For example:

CREATE USER jward
IDENTIFIED BY password
DEFAULT TABLESPACE example
QUOTA 10M ON example
TEMPORARY TABLESPACE temp
QUOTA 5M ON system;

This example creates a local user account and specifies the user password, default tablespace, temporary tablespace where temporary segments are created, tablespace quotas.

2. At minimum, grant the user the CREATE SESSION privilege so that the user can access the database instance.

   GRANT CREATE SESSION TO jward;

   NOTE:
   A newly created user cannot connect to the database until he or she has the CREATE SESSION privilege.
   If the user must access Oracle Enterprise Manager, then you should also grant the user the SELECT ANY DICTIONARY privilege.

This section provides instructions for creating and managing user accounts for the people and applications that use your database.

- Viewing User Accounts
- Creating a User Account
- Creating a New User Account by Duplicating an Existing User Account
- Granting Privileges and Roles to a User Account
- Assigning a Tablespace Quota to a User Account
- Modifying a User Account
- Locking and Unlocking User Accounts
- Expiring a User Password
- Deleting a User Account

Viewing User Accounts

You view user accounts on the Users page of Oracle Enterprise Manager Database Control (Database Control) and from CLI (Command Line Interface) via Putty.

SQL> select username, account_status, default_tablespace from dba_users;

Creating a User Account

Suppose you want to create a user account for a database application developer named Nick. Because Nick is a developer, you want to grant him the database privileges and roles that he requires to build and test his applications. You also want to give Nick a 10 MB quota on his default tablespace so that he can create schema objects in that tablespace.

To Create new user

SQL> create user usr1 identified by usr1;

To create new user by assigning a default tablespace

SQL> create user usr2 identified by usr2 default tablespace users;

Granting Privileges and Roles to a User Account

Suppose you are creating or modifying a user account named Nick. Because Nick is a database application developer, you want to grant him the `APPDEV` role, which enables him to create database objects in his own schema. Because you want Nick to be able to create tables and views in other schemas besides his own, you want to grant him the `CREATE ANY TABLE` and `CREATE ANY VIEW` system privileges. In addition, because he is developing a human resources application, you want him to be able to view the tables in the `hr` sample schema and use them as examples. You therefore want to grant him the `SELECT` object privilege on those tables. Finally, you want Nick to be able to log in to Database Control so that he can use the graphical user interface to create and manage his database objects. You therefore want to grant him the `SELECT_CATALOG_ROLE` role. The following table summarizes the privileges and roles to grant to Nick.

Grant Type	Privilege or Role Name
System privileges	`CREATE ANY TABLE`, `CREATE ANY VIEW`
Object privileges	`SELECT` on all tables in the `hr` schema
Roles	`APPDEV`, `SELECT_CATALOG_ROLE`

Check Database Default Tablespace

When you create a new user without specifying a default tablespace, database default tablespace is assigned to the user.
Use below command to find database default tablespace:

SQL> select PROPERTY_NAME, PROPERTY_VALUE from database_properties where PROPERTY_NAME like '%DEFAULT%';

Modifying a User Account

Suppose you want to remove the quota limitations for the user Nick on his default tablespace, USERS. To do so, you must modify his user account.

Change User Default Tablespace

Use below command to change default tablespace of a user

SQL> alter user usr1 default tablespace example;

Note: The objects created in the old tablespace remain unchanged even after changing a default tablespace for a user

Locking and Unlocking User Accounts

To temporarily deny access to the database for a particular user account, you can lock the user account. If the user then attempts to connect, then the database displays an error message and does not allow the connection.

You can unlock the user account when you want to permit database access again for that user.

To Lock / Unlock user

SQL> alter user usr1 account unlock;
SQL> alter user usr1 account lock;

Expiring a User Password

When you expire a user password, the user is prompted to change his or her password the next time that user logs in.

Reasons to expire a password include the following:

- A user password becomes compromised.
- You have a security policy in place that requires users to change their passwords on a regular basis.
- A user has forgotten his or her password.

About Password Policies

When you create a user account, a default password policy is assigned to that user account. The default password policy for a newly installed database includes these directives:

- The password for the user account expires automatically in 180 days.
- The user account is locked 7 days after password expiration.
- The user account is locked for 1 day after 10 failed login attempts.

For better database security, you may want to impose a more strict password policy. For example, you may want passwords to expire every 70 days, and you may want to lock user accounts after three failed login attempts. (A failed login attempt for a user account occurs when a user enters an incorrect password for the account.)

You may also want to require that passwords be complex enough to provide reasonable protection against intruders who try to break into the system by guessing passwords. For example, you might specify that passwords must contain at least one number and one punctuation mark.

You change the password policy for every user account in the database by modifying the password-related attributes of the DEFAULT profile.

To check current user

SQL> show user

About Assigning a Tablespace Quota for a User

You can assign each user a tablespace quota for any tablespace, except a temporary tablespace.

Assigning a quota accomplishes the following:

• Users with privileges to create certain types of objects can create those objects in the specified tablespace.
• Oracle Database limits the amount of space that can be allocated for storage of a user's objects within the specified tablespace to the amount of the quota.

By default, a user has no quota on any tablespace in the database. If the user has the privilege to create a schema object, then you must assign a quota to allow the user to create objects.

The maximum amount of space that you can assign for a tablespace is 2 TB. If you need more space, then specify UNLIMITED for the QUOTA clause.

You can assign a user either individual quotas for a specific amount of disk space in each tablespace or an unlimited amount of disk space in all tablespaces. Specific quotas prevent a user's objects from using too much space in the database.

You can assign quotas to a user tablespace when you create the user, or add or change quotas later. (You can find existing user quotas by querying the USER_TS_QUOTAS view.)

CREATE USER Statement for Assigning a Tablespace Quota

The QUOTA clause of the CREATE USER statement assigns the quotas for a tablespace.

The following CREATE USER statement assigns quotas for the test_ts and data_ts tablespaces:

CREATE USER jward IDENTIFIED BY password DEFAULT TABLESPACE data_ts QUOTA 500K ON data_ts QUOTA 100M ON test_ts TEMPORARY TABLESPACE temp_ts PROFILE clerk CONTAINER = CURRENT;

You can also assign a quota of `UNLIMITED`.

Tablespace Quota
You can specify a limit onto how much tablespace quota (size) a user can use

SQL> Alter user usr1 quota 10M on users;

Note: Allocating quota doesn't represent reserving the space.
If 2 or more users are sharing a tablespace, quota will be filled up in first come first serve basis

Grants to Users for the UNLIMITED TABLESPACE System Privilege

To permit a user to use an unlimited amount of any tablespace in the database, grant the user the UNLIMITED TABLESPACE system privilege.

The UNLIMITED TABLESPACE privilege overrides all explicit tablespace quotas for the user. If you later revoke the privilege, then you must explicitly grant quotas to individual tablespaces. You can grant this privilege only to users, not to roles.

Before granting the UNLIMITED TABLESPACE system privilege, consider the consequences of doing so.

Advantage:
• You can grant a user unlimited access to all tablespaces of a database with one statement.

Disadvantages:
• The privilege overrides all explicit tablespace quotas for the user.
• You cannot selectively revoke tablespace access from a user with the UNLIMITED TABLESPACE privilege. You can grant selective or restricted access only after revoking the privilege.

TEMPORARY TABLESPACE Clause for Assigning a Temporary Tablespace

The TEMPORARY TABLESPACE clause in the CREATE USER statement assigns a user a temporary tablespace.

CREATE USER usr1 IDENTIFIED BY password DEFAULT TABLESPACE data_ts QUOTA 100M ON test_ts QUOTA 500K ON data_ts TEMPORARY TABLESPACE temp_ts PROFILE clerk CONTAINER = CURRENT;

Dropping a User Whose Schema Contains Objects

Deleting a User Account

Suppose user Nick has moved to another department. Because it is no longer necessary for him to have access to the database, you want to delete his user account.
You must use caution when deciding to deleting a user account, because this action also deletes all schema objects owned by the user.

NOTE: To prevent a user from logging in to the database while keeping the schema objects intact, lock the user account instead.

Before you drop a user whose schema contains objects, carefully investigate the implications of dropping these schema objects.

1. Query the DBA_OBJECTS data dictionary view to find the objects that are owned by the user.

For example:
SQL> SELECT OWNER, OBJECT_NAME FROM DBA_OBJECTS WHERE OWNER LIKE 'ANDY';

Enter the user name in capital letters. Pay attention to any unknown cascading effects. For example, if you intend to drop a user who owns a table, then check whether any views or procedures depend on that particular table.

2. Use the DROP USER SQL statement with the CASCADE clause to drop the user and all associated objects and foreign keys that depend on the tables that the user owns.

Administrative accounts and privileges enable you to perform administrative functions such as:

- Managing users
- Managing database memory
- Starting up and shutting down the database.

This section contains the following topics:

- SYS and SYSTEM Users
- SYSDBA and SYSOPER System Privileges

SYS

- This account can perform all administrative functions.
- All base (underlying) tables and views for the database data dictionary are stored in the SYS schema.
- They should never be modified by any user or database administrator.
- You must not create any tables in the SYS schema.
  The SYS user is granted the SYSDBA privilege, which enables a user to perform high-level administrative tasks such as backup and recovery.

SYSTEM

This account can perform all administrative functions except the following:

- Backup and recovery
- Database upgrade

SYSDBA

- Perform STARTUP and SHUTDOWN operations
- ALTER DATABASE: open, mount, backup, or change character set
- CREATE DATABASE
- DROP DATABASE
- CREATE SPFILE
- ALTER DATABASE ARCHIVELOG
- ALTER DATABASE RECOVER
- Includes the RESTRICTED SESSION privilege

This administrative privilege allows most operations, including the ability to view user data. It is the most powerful administrative privilege.

SYSOPER

- Perform STARTUP and SHUTDOWN operations
- CREATE SPFILE
- ALTER DATABASE: open, mount, or back up
- ALTER DATABASE ARCHIVELOG
- ALTER DATABASE RECOVER
- (Complete recovery only. Any form of incomplete recovery, such as UNTIL TIME|CHANGE|CANCEL|CONTROLFILE requires connecting as SYSDBA.)
- Includes the RESTRICTED SESSION privilege

This privilege allows a user to perform basic operational tasks, but without the ability to view user data.

ORAPWD Utility for Changing the SYS User Password

The ORAPWD utility enables you to change the SYS user password.

You can use the ORAPWD utility with the INPUT_FILE parameter to change the SYS user password. To migrate the password files to a specific format, include the FORMAT option. By default, the format is 12.2 if you do not specify the FORMAT option.

To set a new password for the SYS user using the ORAPWD utility, set the SYS option to Y (yes), use the INPUT_FILE parameter to specify the current password file name, and use the FILE parameter to create the password file to which the original password file is migrated.

For example:

ORAPWD INPUT_FILE='orapworcl' FILE='orapwd' SYS=Y
Enter password for SYS: new_password

Replace new_password with a password that is secure. If you do not want to migrate the password file to a different format, then you can specify the same format as the input_file.

For example, assuming that the input file orapworcl format is 12 and you want to change the SYS user password:

ORAPWD INPUT_FILE='orapworcl' FILE='orapwd'
FORMAT=12 SYS=Y
Enter password for SYS: new_password