

Profiles

A profile is a collection of attributes that apply to a user.

The profile is used to enable a single point of reference for multiple users who share these attributes.

You should assign a profile to each user. Each user can have only one profile.

Profile resource limits are enforced only when you enable resource limitation for the associated database. Enabling this limitation can occur either before starting the database (using the RESOURCE_LIMIT initialization parameter) or while it is open (using the ALTER SYSTEM statement)

Prerequisites

To create a profile, you must have the CREATE PROFILE system privilege.

To specify resource limits for a user, you must:

- Enable resource limits dynamically with the ALTER SYSTEM statement or with the initialization parameter RESOURCE_LIMIT. This parameter does not apply to password resources. Password resources are always enabled.
- Create a profile that defines the limits using the CREATE PROFILE statement
- Assign the profile to the user using the CREATE USER or ALTER USER statement

To find the default profile values, you can run the following query:

```
SQL> SELECT * FROM DBA_PROFILES WHERE PROFILE = 'DEFAULT';
```

PROFILE		RESOURCE_NAME	RESOURCE_TYPE LIMIT
DEFAULT		COMPOSITE_LIMIT KERNEL	UNLIMITED
DEFAULT		SESSIONS_PER_USER KERNEL	UNLIMITED
DEFAULT		CPU_PER_SESSION KERNEL	UNLIMITED
DEFAULT		CPU_PER_CALL KERNEL	UNLIMITED
DEFAULT		LOGICAL_READS_PER_SESSION KERNEL	UNLIMITED
DEFAULT		LOGICAL_READS_PER_CALL KERNEL	UNLIMITED
DEFAULT		IDLE_TIME KERNEL	UNLIMITED
DEFAULT		CONNECT_TIME KERNEL	UNLIMITED
DEFAULT		PRIVATE_SGA KERNEL	UNLIMITED
DEFAULT		FAILED_LOGIN_ATTEMPTS PASSWORD	10

DEFAULT		PASSWORD_LIFE_TIME PASSWORD	180
DEFAULT		PASSWORD_REUSE_TIME PASSWORD	UNLIMITED
DEFAULT		PASSWORD_REUSE_MAX PASSWORD	UNLIMITED
DEFAULT		PASSWORD_VERIFY_FUNCTION PASSWORD	NULL
DEFAULT		PASSWORD_LOCK_TIME PASSWORD	1
DEFAULT		PASSWORD_GRACE_TIME PASSWORD	7

Creating a Profile

- A profile can encompass limits for a specific category, such as limits on passwords or limits on resources.
- To create a profile, you must have the CREATE PROFILE system privilege.
- To find all existing profiles, you can query the DBA_PROFILES view.

For example, to create a profile that defines **password limits**:

```
CREATE PROFILE password_prof
LIMIT
FAILED_LOGIN_ATTEMPTS 2
PASSWORD_LIFE_TIME 60 PASSWORD_REUSE_TIME 60
PASSWORD_REUSE_MAX 5
PASSWORD_LOCK_TIME 1/24
PASSWORD_GRACE_TIME 10
PASSWORD_VERIFY_FUNCTION DEFAULT;
```

Note: **password lock time by default is for 1 day. You can specify it in minutes (n/1440) or even in seconds (n/86400)**

The following example shows how to create a **resource limits profile**.

```
CREATE PROFILE app_user
LIMIT
SESSIONS_PER_USER UNLIMITED
CPU_PER_SESSION UNLIMITED
CPU_PER_CALL 3500
CONNECT_TIME 50
LOGICAL_READS_PER_SESSION DEFAULT
LOGICAL_READS_PER_CALL 1200
PRIVATE_SGA 20K
COMPOSITE_LIMIT 7000000;
```

If you assign the `app_user` profile to a user, the user is subject to the following limits in subsequent sessions:

- The user can have any number of concurrent sessions.
- In a single session, the user can consume an unlimited amount of CPU time.
- A single call made by the user cannot consume more than 35 seconds of CPU time.
- A single session cannot last for more than 45 minutes.
- In a single session, the number of data blocks read from memory and disk is subject to the limit specified in the **DEFAULT** profile.
- A single call made by the user cannot read more than 1200 data blocks from memory and disk.
- A single session cannot allocate more than 20 kilobytes of memory in the SGA.
- In a single session, the total resource cost cannot exceed 7 million service units. The formula for calculating the total resource cost is specified by the **ALTER RESOURCE COST** statement.
- Since the **app_user** profile omits a limit for **IDLE_TIME** and for password limits, the user is subject to the limits on these resources specified in the **DEFAULT** profile.

Assigning a Profile to a User

After you create a profile, you can assign it to users.

You can assign a profile to a user who has already been assigned a profile, but the most recently assigned profile takes precedence. When you assign a profile to an external user or a global user, the password parameters do not take effect for that user.

To find the profiles that are currently assigned to users, you can query the **DBA_USERS** view.

- Use the **ALTER USER** statement to assign the profile to a user.

For example:

```
ALTER USER usr2 PROFILE password_prof;
```

Dropping Profiles

You can drop a profile, even if it is currently assigned to a user.

When you drop a profile, the drop does not affect currently active sessions. Only sessions that were created after a profile is dropped use the modified profile assignments. To drop a profile, you must have the **DROP PROFILE** system privilege.

NOTE:

You cannot drop the default profile.

Any user currently assigned to a profile that is dropped is automatically assigned to the **DEFAULT profile. The **DEFAULT** profile cannot be dropped.**

- Use the SQL statement **DROP PROFILE** to drop a profile. To drop a profile that is currently assigned to a user, use the **CASCADE** option.

For example:

```
DROP PROFILE clerk CASCADE;
```

Query to View Memory Use for Each User Session

The V\$SESSION dynamic view lists the memory use for each user session.

The following query lists all current sessions, showing the Oracle Database user and current User Global Area (UGA) memory use for each session:

```
SELECT USERNAME, VALUE || 'bytes' "Current UGA memory" FROM V$SESSION sess, V$SESSTAT stat,  
V$STATNAME name WHERE sess.SID = stat.SID AND stat.STATISTIC# = name.STATISTIC# AND  
name.NAME = 'session uga memory';
```

Note: Resource management parameters will take in effect only if RESOURCE_LIMIT parameter is set to TRUE.

Use below command to check the RESOURCE_LIMIT parameter

```
SQL> show parameter resource_limit;
```

By default the parameter is set to FALSE. You can change it via below

```
SQL> alter system set resource_limit=TRUE scope=both;
```

To check profiles assigned to a user

```
SQL> SELECT USERNAME, PROFILE FROM DBA_USERS WHERE USERNAME='USR1';
```

To check profile parameter values

```
SQL> SELECT * FROM DBA_PROFILES WHERE PROFILE='&PROFILE_NAME';
```