Lab 5: Access Control

Due date: Thursday, May 14, 12:00pm (noon).

Lab Overview

The lab is designed to give you some hands-on experience with SQL's `GRANT` and `REVOKE` commands. The tasks require access to three user accounts, and therefore, are best done in groups of three. If your group has fewer than three people, talk to the instructor, temporary access to an extra Oracle account will be arranged.

The lab's duration is one lab period and it is due at the end of the period (plus a grace period to print the submission).

Assignment

Each group is expected to have access to three Oracle accounts. These accounts will be referred to as User1, User2 and User3.

The lab consists of a number of assignments. Each assignment asks a question related to database access control rules. The goal of each team is to investigate the answer to the question. To answer each question you will have to use `GRANT` command to grant access to some database objects from one user to another, as well as `REVOKE` command to suspend such access.

Submission materials

For each team, only one submission is required. For each task, you have to submit a document called combined activity log.

It is expected that each task will involve issuing SQL commands and observing their results from multiple user accounts. The easy way to do so is to have multiple sqlplus sessions open, one per user, and issue commands as appropriate. A Combined activity log combines together all commands (and their results) issued from all sqlplus sessions to achieve a specific goal.

A combined activity log is a text file which contains the list of SQL statements and the results of running them. The following rules shall be obeyed:

- SQL commands must be included (cut-and-pasted) into the combined activity log in the order in which they were issued.
- For each SQL command, you have to specify the user account from which the command was issued.

Example. Suppose User1 granted User2 a `SELECT` permission on table `Test`. User2 then executes SELECT * FROM User1.Test; query which returned no rows. The combined activity log for these two statements should look as follows:

```
rem
rem Combined Activity Log. Team members: Alex Dekhtyar, Alex Dekhtyar
rem

rem User1
SQL> GRANT SELECT ON Test TO User2;

Grant succeeded.

SQL>

rem User2

SQL> Select * from User1.Test;

now rows selected
```

Tasks

Task 1. User1 owns table `R1(A int primary key, B int)`. User2 owns table R2(B int primary key, C int). User3 owns table R3(A int, B int, C int).

Suppose User2 wants to compute the natural join of `R2` and `R3` and put all the results into `R1`. Create the permission structure enabling this operation, and show that this operation succeeds. (you will also need SQL commands to create tables and to put some data into the tables).

Submit combined log of your activities.

Task 2. Permission transfer via GRANT OPTION. Users can grant each other access to data via `GRANT` command. A user can also delegate to another user the ability to grant access to the data using the `WITH GRANT OPTION` clause of the `GRANT` command. Suppose User1 gave User2 access to data with the grant option. Suppose User2 used the grant option to grant access to the same data to User3.

Investigate what happens to the structure of permissions when User1 revokes grant option from User2. Test if User3 can keep his/her grant option, if is has been obtained from User2.

Submit combined activity log and a short description of your findings.

Task 3. Mandatory Access Control simulation. Simulate mandatory access control on a single relational table `Reports(A int Primary Key, B int, C CHAR(10))`. User1 has clearance level C(ommon), User2 has clearance level S(ecret) and User3 has clearacne level TS (top secret). $C < S < TS$. All contents of the `Reports` table have classification level S.

You need to create the table and assign appropriate access permissions. Use `SELECT` as a proxy for read access and `INSERT` as a proxy for write access. You also need to demonstrate that your permission assignment correctly implements mandatory access control by running appropriate SQL statements and submitting their results.

Submit the combined activity log for the task.

Submission

Your deliverable is the combined activity logs for each task. The soft copy of the logs has to be submitted to the instructor via `handin` as a single `.zip` or a `.tar.gz` archive:

```
handin dekhtyar lab05-366 <file>
```

You also need to submit the hard copy of your combined activities log. If you finish after the lab is over, bring the printout to my office.