**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**


# BÁO CÁO ĐỒ ÁN MÔN HỌC

## CSC12001 - AN TOÀN VÀ BẢO MẬT DỮ LIỆU TRONG HTTT

**Giáo viên hướng dẫn:** TS. Phạm Thị Bách Huệ

ThS. Lương Vĩ Minh

ThS. Tiết Gia Hồng

**Nhóm sinh viên thực hiện:** ATBM-CQ-08

21120439 - Bùi Minh Duy

21120485 - Hoàng Thị Khôn

21120555 - Nguyễn Hữu Thắng

21120582 - Đinh Hoàng Trung

*Thành phố Hồ Chí Minh – Tháng 6 năm 2024*

I. THÔNG TIN NHÓM 3
II. DANH SÁCH CHỨC NĂNG ĐÃ HOÀN THÀNH 3
III. PHÂN CÔNG VÀ ĐÁNH GIÁ 4
IV. NỘI DUNG CHÍNH 6
1. PHÂN HỆ 1: HỆ THỐNG DÀNH CHO NGƯỜI QUẢN TRỊ BẢO MẬT 6
1.1. Tạo user ADMIN1 6
1.2. Quản lý user 6
1.3. Quản lý role 10
2. PHÂN HỆ 2: HIỆN THỰC CÁC CHÍNH SÁCH BẢO MẬT 15
2.1. Chính sách điều khiển truy cập (Access Control) 15
2.1.1. Lược đồ cơ sở dữ liệu 15
2.1.2. Phát biểu lại các chính sách 15
2.1.3. Kịch bản cài đặt 17
2.2. OLS 20
2.3. Audit 22
2.4. Sao lưu và phục hồi dữ liệu chủ động. 26
V. TÀI LIỆU THAM KHẢO 32

## I. THÔNG TIN NHÓM

| STT | MSSV | Họ và tên | Chú ý |
|---|---|---|---|
| 1 | 21120439 | Bùi Minh Duy | Trưởng nhóm |
| 2 | 21120485 | Hoàng Thị Khôn | |
| 3 | 21120555 | Nguyễn Hữu Thắng | |
| 4 | 21120582 | Đinh Hoàng Trung | |

## II. DANH SÁCH CHỨC NĂNG ĐÃ HOÀN THÀNH

| Phân hệ | Công việc | UI | PL/SQL |
|---|---|---|---|
| 1 | Xem danh sách các đối tượng hiện có trên CSDL (user, role) | ✓ | ✓ |
| 1 | Thêm mới đối tượng (user, role) | ✓ | ✓ |
| 1 | Phân quyền/ lấy lại quyền của một user/ role | ✓ | ✓ |
| 1 | Xem quyền của một chủ thể cụ thể | ✓ | ✓ |
| 2 | Cài đặt chính sách 1 | ✓ | ✓ |
| 2 | Cài đặt chính sách 2 | ✓ | ✓ |
| 2 | Cài đặt chính sách 3 | ✓ | ✓ |
| 2 | Cài đặt chính sách 4 | ✓ | ✓ |
| 2 | Cài đặt chính sách 5 | ✓ | ✓ |
| 2 | Cài đặt chính sách 6 | ✓ | ✓ |
| 2 | OLS | | ✓ |
| 2 | Audit | ✓ | ✓ |
| 2 | Sao lưu và phục hồi dữ liệu chủ động | | ✓ |

## III. PHÂN CÔNG VÀ ĐÁNH GIÁ

| Phân hệ | Người thực hiện | Công việc | % Hoàn thành |
|---|---|---|---|
| 1 | 21120439<br>Bùi Minh Duy | 1. Cài đặt backend giao diện user, login database<br>2. Quay video và hướng dẫn sử dụng giao diện user | 100% |
| | 21120485<br>Hoàng Thị Khôn | 1. Cài đặt giao diện xem chi tiết role<br>2. Merge code và chỉnh sửa giao diện | 100% |
| | 21120555<br>Nguyễn Hữu Thắng | 1. Cài đặt frontend giao diện user, login database<br>2. Tổng hợp tài liệu nộp | 100% |
| | 21120582<br>Hoàng Đình Trung | 1. Cài đặt giao diện role<br>2. Quay video và hướng dẫn sử dụng giao diện role và xem chi tiết role | 100% |
| 2 | 21120439<br>Bùi Minh Duy | 1. Triển khai chính sách điều khiển truy xuất cho vai trò nhân viên cơ bản, sinh viên<br>2. Triển khai ứng dụng cho vai trò sinh viên<br>3. Triển khai OLS<br>4. Báo cáo và quay video | 100% |
| | 21120485<br>Hoàng Thị Khôn | 1. Triển khai chính sách điều khiển truy xuất cho vai trò trưởng đơn vị, trưởng khoa<br>2. Triển khai ứng dụng cho vai trò trưởng khoa<br>3. Triển khai audit<br>4. Báo cáo và quay video | 100% |
| | 21120555<br>Nguyễn Hữu Thắng | 1. Triển khai chính sách điều khiển truy xuất cho vai trò giáo vụ<br>2. Triển khai ứng dụng cho vai trò giáo vụ<br>3. Tạo cơ sở dữ liệu<br>4. Báo cáo và quay video | 100% |

| | | | |
|---|---|---|---|
| | 21120582<br>Hoàng Đình Trung | 1. Triển khai chính sách điều khiển truy xuất giảng viên<br>2.Triển khai ứng dụng cho vai trò nhân viên cơ bản, giáo viên, trưởng đơn vị<br>3. Sao lưu và phục hồi dữ liệu chủ động<br>4. Báo cáo và quay video | 100% |

# IV. NỘI DUNG CHÍNH

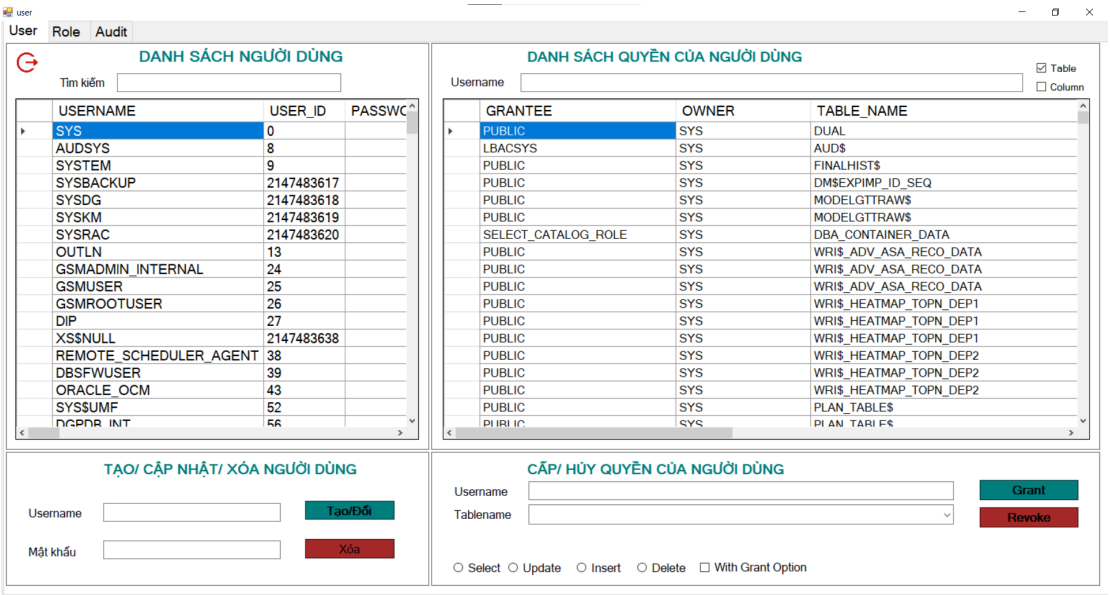
## 1. PHÂN HỆ 1: HỆ THỐNG DÀNH CHO NGƯỜI QUẢN TRỊ BẢO MẬT

### 1.1. Tạo user ADMIN1

Để tạo một user có quyền quản trị trên Oracle DB sử dụng các lệnh SQL sau:

```
CREATE USER ADMIN1 IDENTIFIED BY 123;

GRANT DBA TO ADMIN1;
GRANT EXECUTE ANY PROCEDURE TO ADMIN1;
GRANT CREATE SESSION TO ADMIN1 CONTAINER = ALL;
GRANT CONNECT TO ADMIN1 WITH ADMIN OPTION;
GRANT SELECT ANY DICTIONARY TO ADMIN1;
GRANT CREATE SESSION, CREATE VIEW, ALTER SESSION, CREATE
SEQUENCE TO ADMIN1;
GRANT CREATE SYNONYM, CREATE DATABASE LINK, RESOURCE,
UNLIMITED TABLESPACE TO ADMIN1;
GRANT CREATE USER, CREATE ROLE, ALTER USER, ALTER ANY
ROLE, DROP USER, DROP ANY ROLE TO ADMIN1;
GRANT CREATE TRIGGER TO ADMIN1;
GRANT EXECUTE ON SYS.DBMS_SESSION TO ADMIN1;
GRANT EXECUTE ON DBMS_CRYPTO TO ADMIN1;
```

### 1.2. Quản lý user




Sử dụng các câu lệnh/ thủ tục sau để thực hiện các chức năng liên quan đến quản lý các user trong Oracle Database:

- Xem danh sách user:

```
SELECT * FROM DBA_USERS;
```

- Tạo user mới (có hoặc không có mật khẩu):

```
CREATE USER username {IDENTIFIED BY password}
```

- Cấp quyền connect cho user:

```
GRANT CONNECT TO username
```

- Cập nhập user (đổi mật khẩu):

```
ALTER USER username IDENTIFIED BY newPassword
```

- Xóa role mới:

```
DROP USER username;
```

- Xem danh sách quyền của user trên table và view:

```
SELECT * FROM DBA_TAB_PRIVS;
```

- Xem danh sách quyền của user trên column:

```
SELECT * FROM DBA_COL_PRIVS;
```

- Cấp quyền cho user:

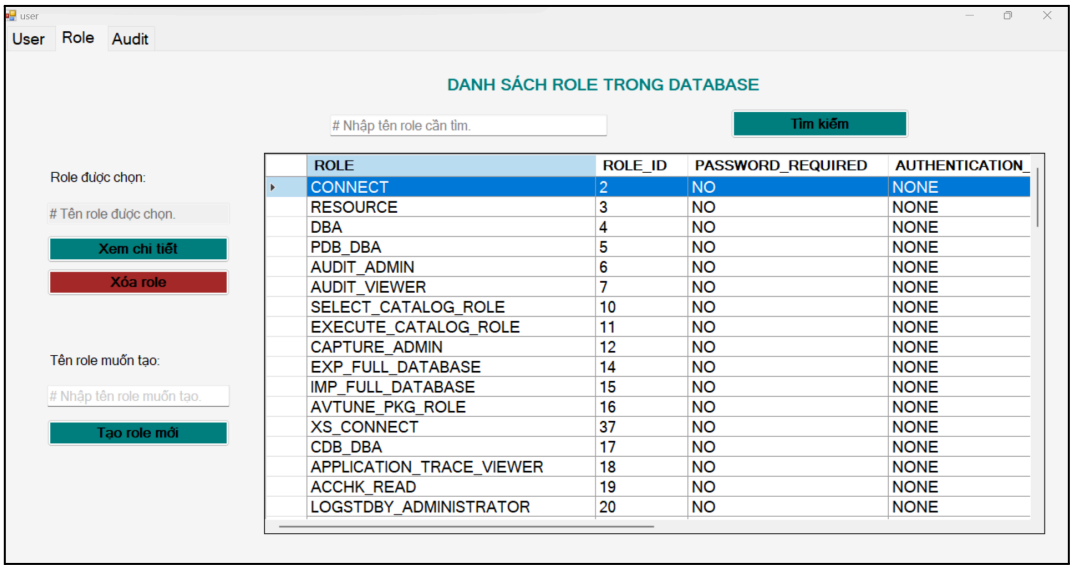
```
create or replace procedure admin1.sp_grant_privilege (
   user_role_name in varchar2,              -- Tên user/role được cấp
quyền
   table_name in varchar2,                  -- Tên bảng
   privilege_type in varchar2,              -- Loại quyền (select, update,
insert, delete)
   column_name in varchar2,                 -- Tên cột (dành cho insert,
update)
   with_grant_option in varchar2 default 'no' -- Tuỳ chọn grant option
)
is
```

```
    strsql varchar2(1000);
    view_name varchar2(1000);
begin
    -- Kiểm tra nếu privilege_type là 'select' và column_name không null,
thì tạo view và cấp quyền
    if privilege_type = 'select' and column_name is not null then
        -- Tạo view từ bảng và cột được chỉ định
        view_name := 'V_' || table_name || '_' || user_role_name ;
        execute immediate 'create or replace view ' || view_name || ' as
select ' || column_name || ' from ' || table_name;

        -- Cấp quyền cho người dùng/role trên view vừa tạo
        strsql := 'grant select on ' || view_name || ' to ' || user_role_name;
        if with_grant_option = 'yes' then
            strsql := strsql || ' with grant option';
        end if;
        execute immediate strsql;
    else
        -- Xây dựng câu lệnh GRANT
        strsql := 'grant ' || privilege_type ||
                  case
                      when column_name is not null then '(' || column_name ||
')'
                      else ''
                  end ||
                  ' on ' || table_name || ' to ' || user_role_name;

        -- Nếu tuỳ chọn "with grant option" được đặt là 'yes', thêm mệnh
đề "with grant option"
        if with_grant_option = 'yes' then
            strsql := strsql || ' with grant option';
        end if;

        -- Thực thi câu lệnh GRANT
        execute immediate strsql;

        -- Hiển thị thông báo khi quyền được cấp thành công
        dbms_output.put_line('Privilege granted successfully.');
    end if;
exception
    when others then
        -- Ném một ngoại lệ tùy chỉnh để thông báo lỗi cho ứng dụng C#
        RAISE_APPLICATION_ERROR(-20001, 'Error granting
privilege: ' || SQLERRM);
end;
/
```

- Thu hồi quyền của user::

```
CREATE OR REPLACE PROCEDURE admin1.sp_revoke_privilege (
   user_role_name IN VARCHAR2,    -- Tên user/role mà quyền sẽ bị
thu hồi
   table_name IN VARCHAR2,        -- Tên bảng
   privilege_type IN VARCHAR2     -- Loại quyền (select, update,
insert, delete)
)
IS
   strsql VARCHAR2(1000);
BEGIN
   -- Xây dựng câu lệnh SQL để thu hồi quyền
   strsql := 'revoke ' || privilege_type ||
             ' on ' || table_name || ' from ' || user_role_name;

   -- Thực thi câu lệnh SQL
   EXECUTE IMMEDIATE strsql;

   -- Hiển thị thông báo
   DBMS_OUTPUT.PUT_LINE('Privilege revoked successfully.');
EXCEPTION
   WHEN OTHERS THEN
      -- Ném một ngoại lệ tùy chỉnh để thông báo lỗi cho ứng dụng C#
      RAISE_APPLICATION_ERROR(-20001, 'Error revoking
privilege: ' || SQLERRM || strsql);
END;
/
```

## 1.3. Quản lý role




Sử dụng các câu lệnh/ thủ tục sau để thực hiện các chức năng liên quan đến quản lý các role trong Oracle Database:

- Xem danh sách role:

```
SELECT * FROM DBA_ROLES;
```

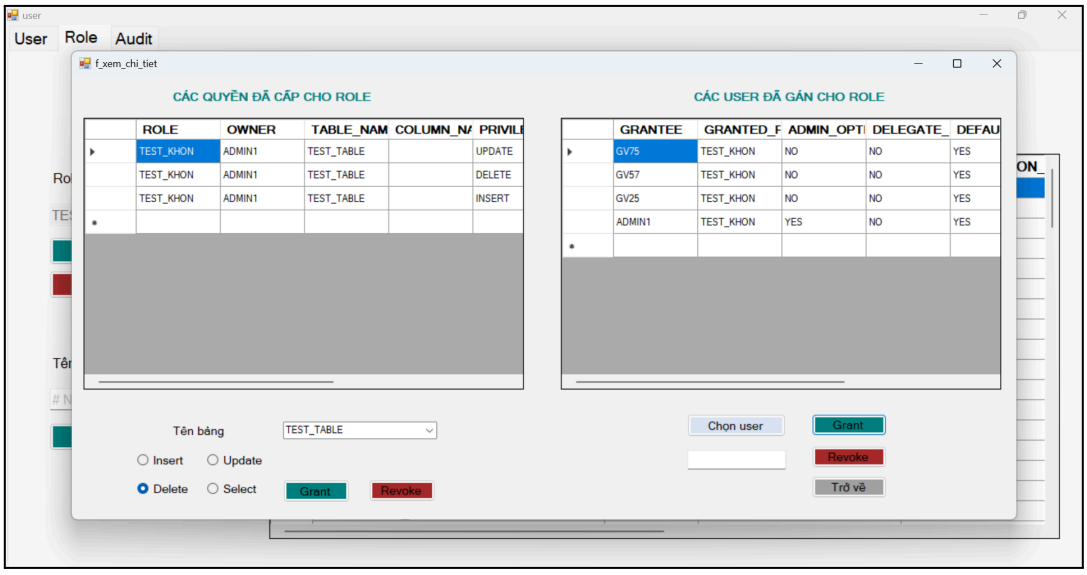
- Tạo role mới:

```
CREATE ROLE rolename;
```

- Xóa role mới:

```
DROP ROLE rolename;
```

❖ Quản lý role cụ thể:




Sử dụng các câu lệnh/ thủ tục sau để thực hiện các chức năng liên quan đến quản lý một role cụ thể trong Oracle Database:

- Xem danh sách quyền của role:

```
SELECT * FROM DBA_ROLE_PRIVS WHERE GRANTED_ROLE = ‘roleName’;
```

- Xem danh sách user đã gán cho role:

```
SELECT * FROM ROLE_TAB_PRIVS WHERE ROLE = 'roleName’;
```

- Cấp quyền cho role:

```
CREATE OR REPLACE PROCEDURE admin1.sp_grant_priv_to_role
(
    p_role_name IN VARCHAR2,
    p_privs IN VARCHAR2,
    p_obj_name IN VARCHAR2,
    p_cols_name IN VARCHAR2 DEFAULT NULL,
    p_error_code OUT NUMBER,
    p_error_msg OUT VARCHAR2
)
IS
    v_sql VARCHAR2(1000);
    v_owner VARCHAR2(50);
    v_view_name VARCHAR2(1000);
BEGIN
    SELECT owner INTO v_owner FROM all_objects WHERE
```

```
object_name = p_obj_name AND ROWNUM = 1;

    IF p_privs = 'Select' AND p_cols_name IS NOT NULL THEN
       -- Phân tách các tên cột và tạo view cho từng cột
       DECLARE
          TYPE col_array IS TABLE OF VARCHAR2(50) INDEX BY
PLS_INTEGER;
          v_cols col_array;
          v_col_name VARCHAR2(50);
       BEGIN
          FOR i IN 1..LENGTH(p_cols_name) -
LENGTH(REPLACE(p_cols_name, ',', '')) + 1 LOOP
             v_col_name := TRIM(REGEXP_SUBSTR(p_cols_name,
'[^,]+', 1, i));
             v_cols(i) := v_col_name;

             -- Tạo view cho từng cột
             v_view_name := 'V_' || p_obj_name || '_' || v_col_name || '_' ||
p_role_name;
             execute immediate 'create or replace view ' || v_view_name || '
as select ' || v_col_name || ' from ' || v_owner || '.' || p_obj_name;

             -- Cấp quyền cho người dùng/role trên view vừa tạo
             v_sql := 'grant select on ' || v_view_name || ' to ' ||
p_role_name;
             execute immediate v_sql;
          END LOOP;
       END;
    ELSE
       IF p_cols_name IS NOT NULL THEN
          v_sql := 'GRANT ' || p_privs || ' (' || p_cols_name || ') ' || ' ON ' ||
v_owner || '.' || p_obj_name || ' TO ' || p_role_name;
       ELSE
          v_sql := 'GRANT ' || p_privs || ' ON ' || v_owner || '.' ||
p_obj_name || ' TO ' || p_role_name;
       END IF;

       EXECUTE IMMEDIATE v_sql;
    END IF;

    p_error_code := 0; -- Success
    p_error_msg := 'Success';

EXCEPTION
    WHEN NO_DATA_FOUND THEN
       p_error_code := -1; -- Custom error code
       p_error_msg := 'No data found for the specified object.';
    WHEN OTHERS THEN
```

```
        p_error_code := SQLCODE;
        p_error_msg := SQLERRM;
END sp_grant_priv_to_role;
/
```

- Thu hồi quyền từ role:

```
-- Thu hồi quyền từ role
CREATE OR REPLACE PROCEDURE
admin1.sp_revoke_priv_from_role (
    p_role_name IN VARCHAR2,
    p_privs IN VARCHAR2,
    p_obj_name IN VARCHAR2,
    p_error_code OUT NUMBER,
    p_error_msg OUT VARCHAR2
)
IS
    v_sql VARCHAR2(1000);
    v_owner VARCHAR2(50);
BEGIN
    SELECT owner INTO v_owner FROM all_objects WHERE
object_name = p_obj_name AND ROWNUM = 1;

    v_sql := 'REVOKE ' || p_privs || ' ON ' ||  v_owner || '.' || p_obj_name
||' FROM ' || p_role_name;

    EXECUTE IMMEDIATE v_sql;
    p_error_code := 0; -- Success
    p_error_msg := 'Success';
    COMMIT;
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            p_error_code := -1; -- Custom error code
            p_error_msg := 'No data found for the specified object.';
        WHEN OTHERS THEN
            p_error_code := SQLCODE;
            p_error_msg := SQLERRM;
END sp_revoke_priv_from_role;
/
```

- Gán role cho user:

```
CREATE OR REPLACE PROCEDURE
admin1.sp_grant_users_to_role (
```

```
    p_user_name IN VARCHAR2,
    p_role_name IN VARCHAR2,
    p_error_code OUT NUMBER,
    p_error_msg OUT VARCHAR2
)
IS
    v_sql VARCHAR2(1000);
    v_count NUMBER;
BEGIN
    v_sql := 'GRANT ' || p_role_name || ' TO ' || p_user_name;

    EXECUTE IMMEDIATE v_sql;
    p_error_code := 0; -- Success
    p_error_msg := 'Success';
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        p_error_code := -1; -- Custom error code
        p_error_msg := 'No data found for the specified object.';
    WHEN OTHERS THEN
        p_error_code := SQLCODE;
        p_error_msg := SQLERRM;
END sp_grant_users_to_role;
/
```

- Thu hồi role đã gán cho user:

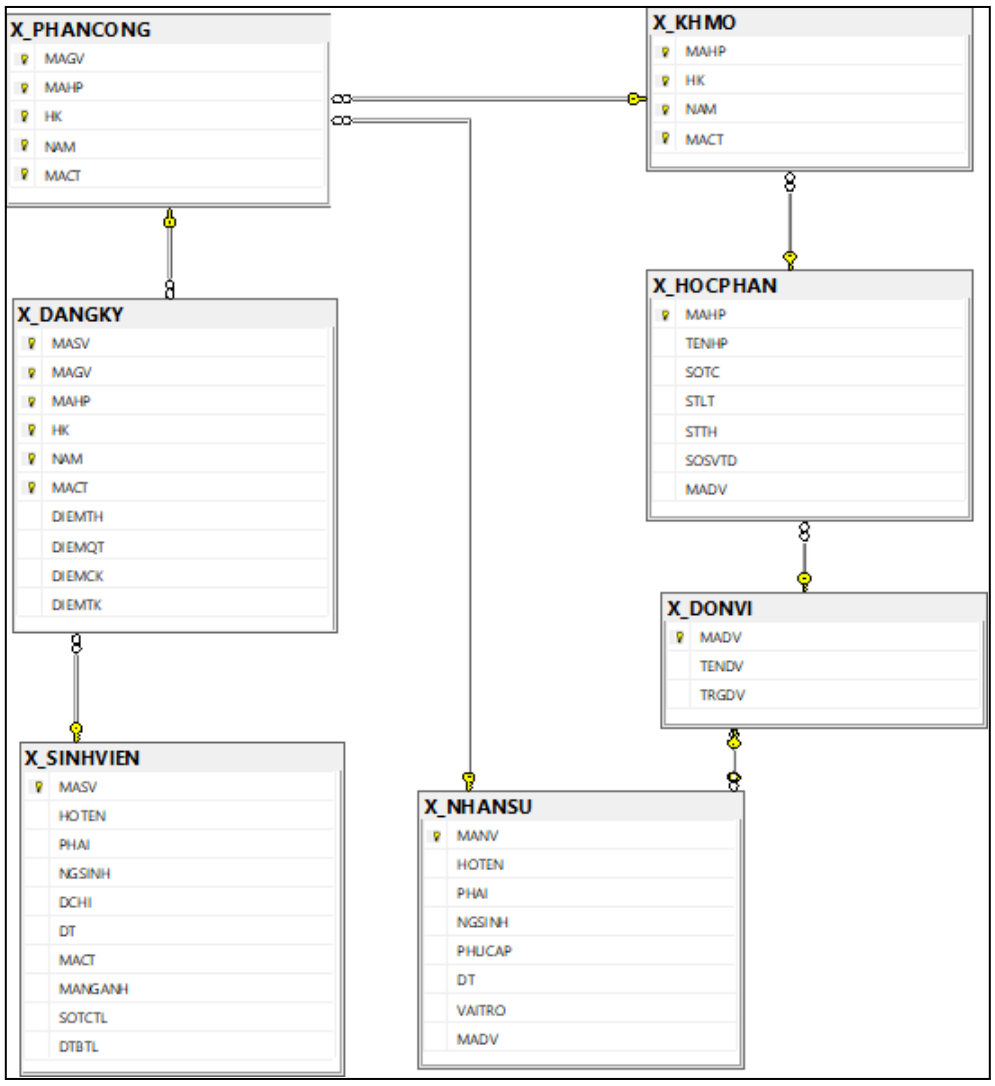
```
CREATE OR REPLACE PROCEDURE
admin1.sp_revoke_user_from_role (
    p_user_name IN VARCHAR2,
    p_role_name IN VARCHAR2,
    p_error_code OUT NUMBER,
    p_error_msg OUT VARCHAR2
)
IS
    v_sql VARCHAR2(1000);

BEGIN
    v_sql := 'REVOKE ' || p_role_name || ' FROM ' || p_user_name;

    EXECUTE IMMEDIATE v_sql;
    p_error_code := 0; -- Success
    p_error_msg := 'Success';
    COMMIT;
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            p_error_code := -1; -- Custom error code
```

```
        p_error_msg := 'No data found for the specified object.';
    WHEN OTHERS THEN
        p_error_code := SQLCODE;
        p_error_msg := SQLERRM;
END sp_revoke_user_from_role;
/
```

# 2. PHÂN HỆ 2: HIỆN THỰC CÁC CHÍNH SÁCH BẢO MẬT

## 2.1. Chính sách điều khiển truy cập (Access Control)

### 2.1.1. Lược đồ cơ sở dữ liệu




### 2.1.2. Phát biểu lại các chính sách

**CS#1:** Người dùng có VAITRO là **"Nhân viên cơ bản"** có quyền truy cập dữ liệu:

- Xem dòng dữ liệu của chính mình trong quan hệ X_NHANSU
- Chỉnh sửa trường DT của dòng dữ liệu của chính mình.

- Xem thông tin của quan hệ X_SINHVIEN, X_DONVI, X_HOCPHAN, X_KHMO.

**CS#2:** Người dùng có VAITRO là **"Giảng viên"** có quyền truy cập dữ liệu:

- Như một người dùng có vai trò "Nhân viên cơ bản" (xem mô tả CS#1).
- Xem dữ liệu phân công giảng dạy liên quan đến bản thân mình trên quan hệ X_PHANCONG.
- Xem dữ liệu trên quan hệ X_DANGKY liên quan đến các lớp học phần mà giảng viên được phân công giảng dạy.
- Cập nhật các trường DIEMTH, DIEMQT, DIEMCK, DIEMTK trên quan hệ X_DANGKY của các sinh viên có tham gia lớp học phần mà giảng viên đó được phân công giảng dạy.

**CS#3:** Người dùng có VAITRO là **"Giáo vụ"** có quyền:

- Như một người dùng có vai trò "Nhân viên cơ bản" (xem mô tả CS#1).
- Xem/ Thêm/ Cập nhật trên các quan hệ X_SINHVIEN, X_DONVI, X_HOCPHAN, X_KHMO theo yêu cầu của trưởng khoa.
- Xem trên quan hệ X_PHANCONG.
- Cập nhật trên quan hệ X_PHANCONG tại các dòng dữ liệu liên quan các học phần do **"Văn phòng khoa"** phụ trách phân công giảng dạy, thừa hành người trưởng đơn vị tương ứng là trưởng khoa.
- Xóa/ Thêm trên quan hệ X_DANGKY theo yêu cầu của sinh viên trong khoảng thời gian còn cho hiệu chỉnh đăng ký, xem điều kiện có thể hiệu chỉnh đăng ký học phần được mô tả bên dưới.

**CS#4:** Người dùng có VAITRO là **"Trưởng đơn vị",** gồm trưởng các bộ môn (không bao gồm trưởng khoa), có quyền truy cập dữ liệu:

- Như một người dùng có vai trò "Giảng viên" (xem mô tả CS#2).
- Thêm/ Xóa/ Cập nhật trên quan hệ X_PHANCONG, đối với các học phần được phụ trách chuyên môn bởi đơn vị mà mình làm trưởng.
- Xem trên quan hệ X_PHANCONG thuộc đơn vị mình làm trưởng

**CS#5:** Người dùng có VAITRO là **"Trưởng khoa"** có quyền hạn:

- Như một người dùng có vai trò "Giảng viên" (xem mô tả CS#2).
- Thêm/ Xóa/ Cập nhật trên quan hệ X_PHANCONG đối với các học phần quản lý bởi đơn vị "Văn phòng khoa".
- Xem/ Thêm/ Xóa/ Cập nhật trên quan hệ X_NHANSU.
- Xem (không giới hạn) dữ liệu trên toàn bộ lược đồ CSDL.

**CS#6:** Người dùng có VAITRO là "Sinh viên" có quyền hạn:

- Xem trên quan hệ X_SINHVIEN tại các dòng dữ liệu của chính mình
- Cập nhật các trường DCHI, DT trên quan hệ X_SINHVIEN các dòng dữ liệu của chính mình.
- Xem trên các quan hệ X_HOCPHAN, X_KHMO của chương trình đào tạo mà sinh viên đang theo học.
- Thêm/ Xóa trên quan hệ X_DANGKY tại các dòng dữ liệu liên quan đến chính sinh viên đó trong học kỳ của năm học hiện tại (nếu thời điểm hiệu chỉnh đăng ký còn hợp lệ).
- **Không** cập nhật các trường DIEMTH, DIEMQT, DIEMCK, DIEMTK trên quan hệ X_DANGKY.
- Xem trên quan hệ X_DANGKY tại các dòng dữ liệu liên quan đến chính sinh viên.

Sinh viên có thể hiệu chỉnh đăng ký học phần (thêm, xóa) nếu ngày hiện tại không vượt quá 14 ngày so với ngày bắt đầu học kỳ (xem thêm thông tin về học kỳ trong quan hệ KHMO) mà sinh viên đang hiệu chỉnh đăng ký học phần.

### 2.1.3. Kịch bản cài đặt

**CS#1:**

- Chủ thể: những người dùng có vai trò "NhanVienCoBan"
- Cơ chế sử dụng: RDAC
- Quyền:
  - X_NHANSU: select, update trên cột (DT)
  - X_SINHVIEN: select
  - X_DONVI: select
  - X_HOCPHAN: select
  - X_KHMO: select
- Kịch bản cài đặt:
  - Tạo vai trò "NhanVienCoBan"
  - Tạo view UV_THONGTINCANHAN_NS để lọc các dòng thông tin trong X_NHANSU liên quan đến chủ thể
  - Tạo procedure USP_UPDATE_SDT_NV để thực hiện update số điện thoại trên view UV_THONGTINCANHAN_NS, cấp quyền thực thi procedure này cho vai trò "NhanVienCoBan"
  - Cấp quyền select, update cột số điện thoại trên view UV_THONGTINCANHAN_NS
  - Cấp quyền select trên bảng X_SINHVIEN, X_DONVI, X_HOCPHAN, X_KHMO
  - Thực hiện cấp vai trò "NhanVienCoBan" cho các người dùng là nhân viên cơ bản.

**CS#2:**

- Chủ thể: những user có là members của role “giangvien” - tức là có vai trò là giảng viên.
- Cơ chế sử dụng: RDAC, VPD.
- Quyền:
    - Các quyền của **CS#1**.
    - X_DANGKY: Select, Update trên cột (DIEMTH, DIEMQT, DIEMCK, DIEMTK)
    - X_PHANCONG: Select.
- Kịch bản cài đặt:
    - Tạo vai trò ‘giangvien’.
    - Cấp vai trò ‘NhanVienCoBan’ cho vai trò ‘giangvien’.
    - Để xem thông tin PHANCONG của bản thân:
        - Tạo View để lọc các dòng thông tin X_PHANCONG liên quan đến chủ thể.
        - Cấp quyền SELECT view vừa mới tạo.
    - Để SELECT, UPDATE trên bảng X_DANGKY:
        - Cài đặt VPD trên bảng X_DANGKY với chính sách cho phép vai trò giangvien select và update các dòng liên quan đến bản thân.
        - Cấp quyền Update trên cột (DIEMTH, DIEMQT, DIEMCK, DIEMTK) của bảng X_DANGKY cho vai trò giangvien.
        - Cấp quyền SELECT trên bảng X_DANGKY cho vai trò giangvien.
    - Thực hiện cấp vai trò ‘giangvien’ cho các user là giảng viên.

**CS#3:**

- Chủ thể: những người dùng có vai trò “RL_GIAOVU” - tức vai trò Giáo vụ
- Cơ chế sử dụng: RDAC
- Quyền:
    - Các quyền của **CS#1**.
    - X_SINHVIEN: select, insert, update
    - X_DONVI: select, insert, update
    - X_HOCPHAN: select, insert, update
    - X_KHMO: select, insert, update
    - X_PHANCONG: select, update (MAGV)
    - X_DANGKY: select, insert, delete
- Kịch bản cài đặt:
    - Tạo vai trò “RL_GIAOVU”
    - Cấp vai trò “NhanVienCoBan” cho vai trò “RL_GIAOVU”.
    - Cấp quyền SELECT, INSERT, UPDATE trên bảng X_SINHVIEN cho vai trò RL_GIAOVU

- Cấp quyền SELECT, INSERT, UPDATE trên bảng X_DONVI cho vai trò RL_GIAOVU
- Cấp quyền SELECT, INSERT, UPDATE trên bảng X_HOCPHAN cho vai trò RL_GIAOVU
- Cấp quyền SELECT, INSERT, UPDATE trên bảng X_KHMO cho vai trò RL_GIAOVU
- Cấp quyền SELECT trên bảng X_PHANCONG cho vai trò RL_GIAOVU
- Tạo procedure USP_UPDATE_PHANCONG để thực hiện cập nhật giáo viên mới cho phân công liên quan đến học phần do Văn phòng khoa phụ trách phân công giảng dạy, cấp quyền thực thi procedure này cho vai trò "RL_GIAOVU"
- Cấp quyền SELECT trên bảng X_DANGKY cho vai trò RL_GIAOVU
- Tạo procedure USP_INSERT_DANGKY để thực hiện việc thêm học phần đăng ký trong khoảng thời gian cho phép hiệu chỉnh, cấp quyền thực thi procedure này cho vai trò "RL_GIAOVU"
- Tạo procedure USP_DELETE_DANGKY để thực hiện việc xóa học phần đăng ký trong khoảng thời gian cho phép hiệu chỉnh, cấp quyền thực thi procedure này cho vai trò "RL_GIAOVU"
- Thực hiện cấp vai trò "RL_GIAOVU" cho các người dùng là giáo vụ.

**CS#4:**

- Chủ thể: những user có là members của role "RL_TRGDV" - tức là có vai trò là trưởng đơn vị.
- Cơ chế sử dụng: VPD, RDAC.
- Quyền:
  - Các quyền của **CS#2**.
  - X_PHANCONG: INSERT, SELECT, UPDATE, DELETE.
- Kịch bản cài đặt:
  - Tạo vai trò 'RL_TRGDV'.
  - Cấp vai trò 'giangvien' cho vai trò 'RL_TRGDV'.
  - Để truy cập được trên X_PHANCONG:
    - Cài đặt VPD trên bảng X_PHANCONG với chính sách cho phép vai trò RL_TRGDV có thể INSERT, SELECT, UPDATE, DELETE các dòng liên quan đến tất cả các giảng viên trong đơn vị chủ thể làm trưởng.
  - Cấp vai trò 'RL_TRGDV ' cho các user là trưởng đơn vị.

**CS#5:**

- Chủ thể: Người dùng "TK" - tức là có vai trò là trưởng khoa.
- Cơ chế sử dụng: DAC, VPD
- Quyền:
  - X_PHANCONG: SELECT, INSERT, UPDATE, DELETE

    - X_NHANSU: SELECT, INSERT, UPDATE, DELETE
    - X_DANGKY: SELECT
- Kịch bản cài đặt:
    - Cấp vai trò GiangVien cho người dùng TK.
    - Thiết lập VPD trên bảng X_PHANCONG với chính sách cho phép trưởng khoa SELECT toàn bộ dữ liệu của bảng X_PHANCONG. Cấp quyền SELECT trên bảng X_PHANCONG.
    - Thiết lập VPD trên bảng X_PHANCONG với chính sách cho phép trưởng khoa chỉ được INSERT, UPDATE, DELETE trên các dòng dữ liệu có MADV là "VPK". Cấp quyền INSERT, UPDATE, DELETE trên bảng X_PHANCONG.
    - Cấp quyền SELECT, INSERT, UPDATE, DELETE trên bảng X_NHANSU.
    - Cấp quyền SELECT trên bảng X_DANGKY.

**CS#6:**

- Chủ thể: những user có là members của role "rl_SinhVien" - tức là nằm trong bảng sinh viên cũng được xem như có vai trò là sinh viên.
- Cơ chế sử dụng: VPD
- Quyền:
    - X_SINHVIEN: select, update(DTH, DCHI)
    - X_KHMO: select
    - X_HOCPHAN: select
    - X_DANGKY: select, insert, delete
- Kịch bản cài đặt:
    - Tạo vai trò 'rl_SinhVien'.
    - Thiết lập VPD trên bảng X_SINHVIEN với chính sách cho phép sinh viên SELECT các dòng dữ liệu của chính mình và UPDATE cột DTH và DCHI trên đó.
    - Thiết lập VPD trên bảng X_KHMO, X_HOCPHAN với chính sách cho phép sinh viên SELECT các dòng dữ liệu của chương trình đào tạo mà sinh viên đang theo học.
    - Thiết lập VPD trên bảng X_DANGKY với chính sách cho phép sinh viên SELECT các dòng dữ liệu liên quan tới chính mình, INSERT và DELETE các dòng dữ liệu liên quan đến chính sinh viên đó trong học kỳ của năm học hiện tại (nếu thời điểm hiệu chỉnh đăng ký còn hợp lệ).
    - Thực hiện cấp vai trò 'rl_SinhVien' cho các user là sinh viên.

## 2.2. OLS

Oracle Label Security (OLS) là một tính năng trong hệ thống quản lý cơ sở dữ liệu Oracle Database. Nó cung cấp các công cụ và khả năng để triển khai và quản lý việc bảo mật dữ liệu trên cấp độ nhãn (label-level) trong hệ thống cơ sở dữ liệu.

OLS cho phép bạn xác định và gắn nhãn cho các đối tượng dữ liệu, chẳng hạn như bảng, cột, dòng, hoặc thậm chí từng giá trị riêng lẻ. Nhãn được sử dụng để đại diện cho 20 các cấp độ bảo mật khác nhau, ví dụ như "cực kỳ bảo mật" (top secret), "bảo mật" (secret), "nội bộ" (internal), và "công khai" (public). Bằng cách gắn nhãn cho dữ liệu, bạn có thể áp dụng các chính sách bảo mật nhằm kiểm soát truy cập dựa trên các quyền và nhãn đã được xác định.

Oracle Label Security hỗ trợ tích hợp với các tính năng khác của Oracle Database như quản lý người dùng và vai trò, quyền hạn, và các công nghệ mã hóa dữ liệu khác. Nó cung cấp khả năng thực hiện kiểm tra kiểm soát truy cập để đảm bảo rằng chỉ những người có quyền được phép xem, sửa đổi, hoặc truy cập vào các đối tượng dữ liệu có nhãn tương ứng.

OLS thường được sử dụng trong các môi trường có yêu cầu bảo mật cao như trong ngành chính phủ, lĩnh vực quân sự, hoặc các tổ chức có nhu cầu bảo vệ dữ liệu nhạy cảm.

**<u>Thiết lập hệ thống nhãn và thực hiện các yêu cầu đề bài:</u>**

Level: TK (Trưởng khoa) > TDV (Trưởng đơn vị) > GV (Giảng viên) > GVu (Giáo vụ) > NV (Nhân viên cơ bản) > SV (Sinh viên)

Compartment: HTTT (Hệ thống thông tin), CNPM (Công nghệ phần mềm), KHMT (Khoa học máy tính), CNTT (Công nghệ thông tin), TGMT (Thị giác máy tính), MMT (Mạng máy tính)

Group: CS1 (Cơ sở 1), CS2 (Cơ sở 2)

a. **Hãy gán nhãn cho người dùng là trưởng khoa có thể đọc được toàn bộ thông báo:** Ta gán cho người dùng *TruongKhoa* nhãn sau

   TK: HTTT, CNPM, KHMT, CNTT, TGMT, MMT: CS1, CS2:

b. **Hãy gán nhãn cho các trưởng bộ môn phụ trách cơ sở 2 có thể đọc được toàn bộ thông báo dành cho trưởng bộ môn không phân biệt vị trí địa lý:** Ta gán cho người dùng *TruongBM_CS2* nhãn sau:

   TDV: HTTT, CNPM, KHMT, CNTT, TGMT, MMT: CS1, CS2

   Hoặc

   *TruongBM_HTTT_CS2* => TDV: HTTT: CS1, CS2

   *TruongBM_CNPM_CS2* => TDV: CNPM: CS1, CS2

   *TruongBM_KHMT_CS2* => TDV: KHMT: CS1, CS2

   *TruongBM_CNTT_CS2* => TDV: CNTT: CS1, CS2

   *TruongBM_TGMT_CS2* => TDV: TGMT: CS1, CS2

   *TruongBM_MMT_CS2* => TDV: MMT: CS1, CS2

c. **Hãy gán nhãn cho 01 giáo vụ có thể đọc toàn bộ thông báo dành cho giáo vụ:** Ta gán cho người dùng *GiaoVu_01* nhãn sau:

   GVu: HTTT, CNPM, KHMT, CNTT, TGMT, MMT: CS1, CS2

d. **Hãy cho biết nhãn của dòng thông báo t1 để t1 được phát tán (đọc) bởi tất cả trưởng đơn vị:**

   t1: TDV

e. **Hãy cho biết nhãn của dòng thông báo t2 để phát tán t2 đến sinh viên thuộc ngành HTTT học ở *Cơ sở 1*:**

   t2: SV: HTTT: CS1

f. **Hãy cho biết nhãn của dòng thông báo t3 để phát tán t3 đến trưởng bộ môn KHMT ở *Cơ sở 1*:**

   t3: TDV: KHMT: CS1

g. **Cho biết nhãn của dòng thông báo t4 để phát tán t4 đến trưởng bộ môn KHMT ở *Cơ sở 1 và Cơ sở 2*:**

   t4: TDV: KHMT: CS1, CS2

h. **Thêm 3 chính sách:**

   - **Phát tán t5 đến sinh viên thuộc ngành CNPM học ở Cơ sở 2:**

     t5: SV: CNPM: CS2

   - **Phát tán t6 đến Trưởng bộ môn MMT ở Cơ sở 2:**

     t6: TDV: MMT: CS2

   - **Phát tán t7 đến HTTT ở Cơ sở 1:**

     t7: GV: HTTT: CS1

## 2.3. Audit

Auditing là quá trình theo dõi và ghi lại chi tiết các hoạt động thao tác của người dùng vào cơ sở dữ liệu. Trên nền tảng Oracle, người quản trị có thể thiết lập và cấu hình để thực hiện audit trên tất cả các người dùng trong hệ thống, bao gồm cả những người dùng không có tài khoản trong cơ sở dữ liệu. Điều này cho phép người quản trị theo dõi rõ ràng các hành vi và hoạt động thực hiện trên dữ liệu. Audit có thể được tinh chỉnh để bao gồm một số lệnh cụ thể hoặc tập trung vào kiểm tra những vai trò (roles) quan trọng trong hệ thống.

1. **<u>Kích hoạt việc ghi nhật ký hệ thống.</u>**

   Cấu hình audit_trail trong Oracle xác định nơi lưu trữ các bản ghi audit. Có nhiều tùy chọn cho audit_trail, nhưng trong trường hợp này, chúng ta sẽ cấu hình để lưu trữ các bản ghi audit trong cơ sở dữ liệu.

*ALTER SYSTEM SET audit_trail = DB SCOPE = SPFILE;*

Giải thích:

- ALTER SYSTEM SET audit_trail = DB SCOPE = SPFILE; cấu hình hệ thống để ghi nhật ký vào cơ sở dữ liệu.
- DB chỉ định rằng các bản ghi audit sẽ được lưu trữ trong các bảng audit của cơ sở dữ liệu.
- SCOPE = SPFILE chỉ định rằng thay đổi này sẽ được áp dụng khi hệ thống khởi động lại, tức là thay đổi sẽ không có hiệu lực ngay lập tức mà sẽ được áp dụng trong lần khởi động tiếp theo của hệ thống.

Sau khi thay đổi cấu hình audit_trail, chúng ta cần khởi động lại cơ sở dữ liệu để thay đổi có hiệu lực. Dưới đây là các bước để thực hiện việc này:

*SHUTDOWN IMMEDIATE;*

*STARTUP;*

Giải thích:

- SHUTDOWN IMMEDIATE; tắt cơ sở dữ liệu một cách an toàn, đảm bảo tất cả các phiên làm việc hiện tại được hoàn tất trước khi tắt.
- STARTUP; khởi động lại cơ sở dữ liệu, áp dụng tất cả các thay đổi đã được cấu hình trong SPFILE.

2. **Thực hiện ghi nhật ký hệ thống dùng Standard audit: theo dõi hành vi của những user nào trên những đối tượng cụ thể, trên các đối tượng khác nhau (table, view, stored procedure, function), hay chỉ định theo dõi các hành vi hiện thành công hay không thành công.**
   - Ghi nhật ký các hành vi cập nhật view V_DANGKY_GV không thành công

     AUDIT UPDATE ON ADMIN1.V_DANGKY_GV WHENEVER NOT SUCCESSFUL;

   - Ghi nhật ký việc thực thi thủ tục ADMIN1.X_NHANSU

     AUDIT DELETE ON ADMIN1.X_NHANSU BY ACCESS;

   - Ghi nhật ký các phiên login không thành công

     NOAUDIT SESSION WHENEVER NOT SUCCESSFUL;

3. **Thực hiện Fine-grained Audit các tình huống sau và tạo ngữ cảnh để có thể ghi vết được (có dữ liệu ghi vết) các hành vi sau:**
   a. **Hành vi Cập nhật quan hệ DANGKY tại các trường liên quan đến điểm số nhưng người đó không thuộc vai trò Giảng viên.**

      **Bước 1:** Tạo hàm kiểm tra vai trò có phải giảng viên không

```
CREATE OR REPLACE FUNCTION IS_GIANGVIEN RETURN
NUMBER IS
   v_count NUMBER := 0;
BEGIN
   -- Kiểm tra số lượng bản ghi từ câu truy vấn
   SELECT COUNT(*)
   INTO v_count
   FROM DBA_ROLE_PRIVS
   WHERE GRANTED_ROLE = 'GIANGVIEN'
   AND GRANTEE = SYS_CONTEXT('USERENV',
'SESSION_USER');
   -- Trả về 0 không là giảng viên, ngược lại trả về 1 là giảng viên
   IF v_count = 0 THEN
      RETURN 0;
   ELSE
      RETURN 1;
   END IF;
END;
/
```

**Bước 2:** Thêm audit trên quan hệ X_DANGKY với chính sách như sau:

```
BEGIN
   DBMS_FGA.ADD_POLICY (
      OBJECT_SCHEMA   => 'ADMIN1',
      OBJECT_NAME     => 'V_DANGKY_GV',
      POLICY_NAME     => 'AUDIT_UPDATE_DIEM',
      AUDIT_CONDITION => 'CHECK_GIANGVIEN() = 0',
      AUDIT_COLUMN    => 'DIEMTH, DIEMQT, DIEMCK, DIEMTK',
      STATEMENT_TYPES => 'UPDATE',
      ENABLE          => TRUE,
```

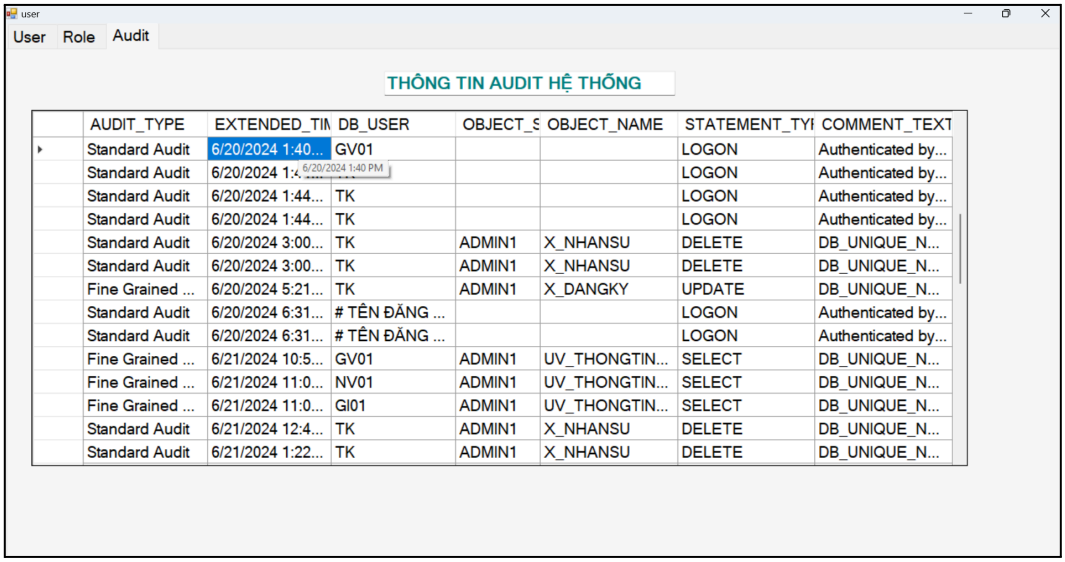
```
    AUDIT_TRAIL     => DBMS_FGA.DB +
DBMS_FGA.EXTENDED
  );
END;
/
```

**b. Hành vi của người dùng này có thể đọc trên trường PHUCAP của người khác ở quan hệ NHANSU.**

Thêm audit trên quan hệ X_NHANSU với chính sách như sau:

```
BEGIN
  DBMS_FGA.ADD_POLICY (
    OBJECT_SCHEMA   => 'ADMIN1',
    OBJECT_NAME     => 'UV_THONGTINCANHAN_NS',
    POLICY_NAME     => 'AUDIT_SELECT_PHUCAP',
    AUDIT_CONDITION => 'MANV = USER',
    AUDIT_COLUMN    => 'PHUCAP',
    STATEMENT_TYPES => 'SELECT',
    ENABLE          => TRUE,
    AUDIT_TRAIL     => DBMS_FGA.DB +
DBMS_FGA.EXTENDED
  );
END;
/
```

4. **Kiểm tra (đọc xuất) dữ liệu nhật ký hệ thống.**




| AUDIT_TYPE | EXTENDED_TIM | DB_USER | OBJECT_S | OBJECT_NAME | STATEMENT_TYI | COMMENT_TEXT |
|---|---|---|---|---|---|---|
| Standard Audit | 6/20/2024 1:40... | GV01 | | | LOGON | Authenticated by... |
| Standard Audit | 6/20/2024 1:... | | | | LOGON | Authenticated by... |
| Standard Audit | 6/20/2024 1:44... | TK | | | LOGON | Authenticated by... |
| Standard Audit | 6/20/2024 1:44... | TK | | | LOGON | Authenticated by... |
| Standard Audit | 6/20/2024 3:00... | TK | ADMIN1 | X_NHANSU | DELETE | DB_UNIQUE_N... |
| Standard Audit | 6/20/2024 3:00... | TK | ADMIN1 | X_NHANSU | DELETE | DB_UNIQUE_N... |
| Fine Grained ... | 6/20/2024 5:21... | TK | ADMIN1 | X_DANGKY | UPDATE | DB_UNIQUE_N... |
| Standard Audit | 6/20/2024 6:31... | # TÊN ĐĂNG ... | | | LOGON | Authenticated by... |
| Standard Audit | 6/20/2024 6:31... | # TÊN ĐĂNG ... | | | LOGON | Authenticated by... |
| Fine Grained ... | 6/21/2024 10:5... | GV01 | ADMIN1 | UV_THONGTIN... | SELECT | DB_UNIQUE_N... |
| Fine Grained ... | 6/21/2024 11:0... | NV01 | ADMIN1 | UV_THONGTIN... | SELECT | DB_UNIQUE_N... |
| Fine Grained ... | 6/21/2024 11:0... | GI01 | ADMIN1 | UV_THONGTIN... | SELECT | DB_UNIQUE_N... |
| Standard Audit | 6/21/2024 12:4... | TK | ADMIN1 | X_NHANSU | DELETE | DB_UNIQUE_N... |
| Standard Audit | 6/21/2024 1:22... | TK | ADMIN1 | X_NHANSU | DELETE | DB_UNIQUE_N... |

Để tra dữ liệu nhật ký hệ thông sử dụng câu lệnh sau:

```
SELECT AUDIT_TYPE, EXTENDED_TIMESTAMP, DB_USER,
OBJECT_SCHEMA, OBJECT_NAME, STATEMENT_TYPE,
COMMENT_TEXT

FROM DBA_COMMON_AUDIT_TRAIL

ORDER BY EXTENDED_TIMESTAMP;
```

## 2.4. Sao lưu và phục hồi dữ liệu chủ động.

### 2.4.1. Sao lưu.

- Trong cơ sở dữ liệu Oracle, "backup" (sao lưu) là quá trình tạo ra một bản sao của dữ liệu từ cơ sở dữ liệu, giúp bảo vệ dữ liệu trước các sự cố như hỏng hóc phần cứng, lỗi phần mềm, lỗi người dùng hoặc các sự cố khác. Mục tiêu của việc sao lưu là đảm bảo rằng dữ liệu có thể được khôi phục lại trạng thái ban đầu hoặc trạng thái gần nhất có thể trong trường hợp xảy ra sự cố.
- Oracle hỗ trợ nhiều loại sao lưu khác nhau, bao gồm:
    - **Full Backup (Sao lưu toàn bộ)**: Sao lưu toàn bộ cơ sở dữ liệu, bao gồm tất cả các tập tin dữ liệu và các tập tin cần thiết khác.
    - **Incremental Backup (Sao lưu gia tăng)**: Sao lưu chỉ những thay đổi kể từ lần sao lưu gần nhất. Điều này có thể giảm thiểu thời gian và không gian lưu trữ cần thiết so với sao lưu toàn bộ.
    - **Differential Backup (Sao lưu khác biệt)**: Sao lưu chỉ những thay đổi kể từ lần sao lưu toàn bộ gần nhất.
- Trong đồ án này sẽ sử dụng 2 cách để back up
    - Cách 1 là tạo job nhằm mục đích tự động hóa backup theo bộ lập lịch.
    - Cách 2 là dùng RMAN để chủ động backup.

#### 2.4.1.1. Sao lưu tự động:

- Để có thể tự động hóa việc lập lịch ta cần tạo các job để thực hiện theo lịch trình được đặt sẵn, nhưng việc backup cần phải chạy RMAN script để thực hiện (là một script job). Do đó để tạo được một job tự động backup ta cần một số quyền như sau:
    - Quyền CREATE JOB: Cho phép user hiện hiện tại tạo các công việc trong Oracle Scheduler.
    - Quyền CREATE EXTERNAL JOB: Cho phép user hiện tại tạo các công việc bên ngoài để chạy các tập lệnh shell hoặc các công việc ngoài Oracle.
    - Thông tin xác thực hệ điều hành (OS credentials): Để xác thực và ủy quyền cho script chạy với quyền truy cập của người dùng hệ điều hành. Bạn có thể sử dụng DBMS_CREDENTIAL để tạo và quản lý các thông tin xác thực này. Credential là một phần quan trọng trong việc thực thi script jobs trong Oracle Scheduler vì nó cung cấp thông tin xác thực để kết nối và chạy các tác vụ trên hệ điều hành (OS).
- **Mối quan hệ giữa BACKUP_SCRIPT và SCRIPT JOB:**

  BACKUP_SCRIPT là một kiểu công việc (JOB_TYPE) trong Oracle DBMS_Scheduler. Nó cho phép bạn chạy các kịch bản RMAN (RMAN script) trực tiếp từ Oracle Database mà không cần phải chuyển sang hệ điều hành để thực hiện các lệnh RMAN.

  Các thành phần chính

- JOB_TYPE:
    - JOB_TYPE xác định loại công việc sẽ thực hiện. Khi JOB_TYPE là BACKUP_SCRIPT, nó cho phép bạn chạy một kịch bản RMAN được chỉ định.
    - Điều này giúp đơn giản hóa quy trình sao lưu vì bạn không cần phải viết các shell script riêng để gọi RMAN từ hệ điều hành.
- JOB_ACTION:
    - JOB_ACTION xác định hành động cụ thể mà công việc sẽ thực hiện. Đối với BACKUP_SCRIPT, JOB_ACTION có thể là một kịch bản RMAN được viết trực tuyến (in-line) hoặc là một đường dẫn đầy đủ đến một tập tin chứa kịch bản RMAN trên hệ thống tệp của máy chủ cơ sở dữ liệu.

  Cách hoạt động: Khi bạn tạo một công việc với JOB_TYPE là BACKUP_SCRIPT, Oracle sẽ thực thi kịch bản RMAN được chỉ định trong JOB_ACTION. Điều này có thể được thực hiện theo hai cách:
  => Vì vậy nên bước đầu tiên trước khi tạo các công việc backup ta cần cấp quyền đầy đủ quyền cần thiết cho ADMIN và khai báo credential để có thể chạy Backup_Script. Ở đây ta chỉ chạy local trên máy nên cần tạo credential như sau:

- Cấp quyền:

```
GRANT CREATE JOB TO admin1;
GRANT CREATE EXTERNAL JOB TO admin1;
```

```
GRANT CREATE CREDENTIAL TO admin1;
```

- Thực hiện các lệnh sau trên SQL*PLUS để chuyển LOG_MODE thành ARCHIVELOG:

```
SHUTDOWN IMMEDIATE;
STARTUP MOUNT;
ALTER DATABASE ARCHIVELOG;
ALTER DATABASE OPEN;
```

- Tạo credential:

```
BEGIN
   DBMS_CREDENTIAL.create_credential(
      credential_name => 'admin1_CREDENTIAL',
      -- username và password của localhost trên máy mình
      username => 'user_của_localhost',
      password => 'Mật_khẩu'
   );
END;
/
```

- Trong đồ án này em sẽ lên lịch backup tự động như sau:
    - Các ngày trong tuần: Thực hiện **Incremental backup** vào lúc 0 giờ.
    - Cuối tuần: Thực hiện **Full backup** vào lúc 0 giờ.

#### 2.4.1.1.1. Full backup.

- Tạo program để lưu hành động khi công việc được gọi:

```
begin
   DBMS_SCHEDULER.CREATE_PROGRAM(
      program_name => 'FULL_BACKUP_PROGRAM',
      program_type => 'BACKUP_SCRIPT',
      program_action => q'[connect target /
         RUN {
            BACKUP INCREMENTAL LEVEL 0 DATABASE;
         }]',
      enabled  => TRUE
   );
end;
/
```

- Tạo Schedule (lịch trình) để xác định thời điểm việc backup sẽ diễn ra:

```
begin
   DBMS_SCHEDULER.CREATE_SCHEDULE(
```

```
    schedule_name=> 'Full_backup_schedule',
    start_date=> systimestamp,
    repeat_interval=> 'FREQ=WEEKLY;
                BYDAY=SUN; BYHOUR=0;
                BYMINUTE=0; BYSECOND=0',
    end_date=> null,
    comments => 'Full back up once a week at 0PM in sunday'
  );
end;
/
```

- Tạo công việc backup với hành động và lịch trình cụ thể:

```
BEGIN
  DBMS_SCHEDULER.create_job (
  job_name => 'FULL_BACKUP_JOB',
  program_name => 'FULL_BACKUP_PROGRAM',
  schedule_name => 'FULL_BACKUP_SCHEDULE',
  credential_name => 'admin1_CREDENTIAL',
  enabled => TRUE
  );
END;
/
```

#### 2.4.1.1.2. Incremental backup.

- Tạo program để lưu hành động khi công việc được gọi:

```
begin
  DBMS_SCHEDULER.CREATE_PROGRAM(
    program_name => 'INCREMENTIAL_BACKUP_PROGRAM',
    program_type => 'BACKUP_SCRIPT',
    program_action => q'[connect target /
          RUN {
            BACKUP INCREMENTAL LEVEL 1 DATABASE;
          }]',
    enabled  => TRUE
  );
end;
/
```

- Tạo Schedule (lịch trình) để xác định thời điểm việc backup sẽ diễn ra:

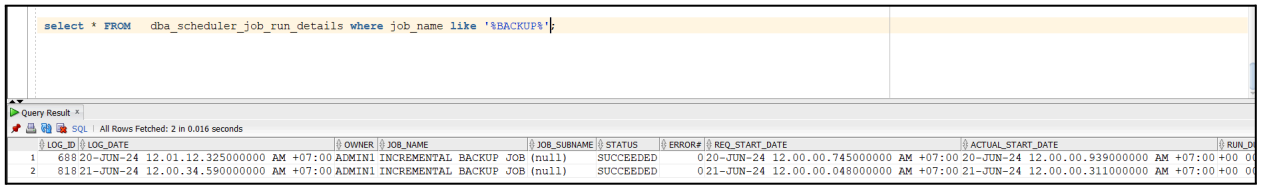
```
BEGIN
```

```
    dbms_scheduler.create_schedule(
        schedule_name=> 'INCREMENTIAL_BACKUP_SCHEDULE',
        start_date=> systimestamp,
        repeat_interval=>   'FREQ=WEEKLY;
                            BYDAY=MON,TUE,WED,THU,FRI,SAT;
                            BYHOUR=0;
                            BYMINUTE=0;
                            BYSECOND=0',
        end_date=> NULL,
        comments => 'Incremental backup at 0 o"clock in weekdays'
    );
END;
/
```

- Tạo công việc backup với hành động và lịch trình cụ thể:

```
Begin
    Dbms_Scheduler.Create_Job (
    Job_Name => 'INCREMENTAL_BACKUP_JOB',
    Program_Name => 'INCREMENTAL_BACKUP_PROGRAM',

    Schedule_Name =>   'INCREMENTAL_BACKUP_SCHEDULE',
    Credential_Name => 'admin1_CREDENTIAL',
    Enabled => True
    );
End;
/
```

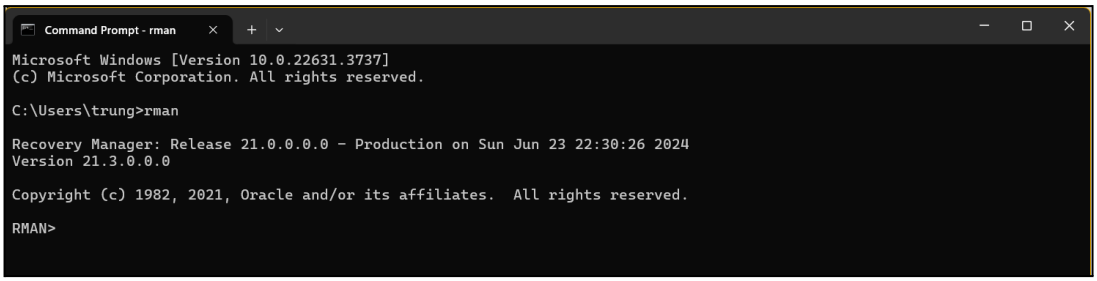
- Để xem được chi tiết các lần chạy của backup job ta có thể thực hiện truy vấn vào view ***_scheduler_job_run_details** - với * là:
    - DBA: các job trên toàn bộ database.
    - USER: các job thuộc quyền sở hữu của user hiện tại.
    - ALL: các job mà user hiện tạo có quyền sử dụng.




#### 2.4.1.2. Sao lưu chủ động:

- Việc sao lưu chủ động sẽ được thực hiện bằng RMAN sử dụng trên CMD với một số cú pháp sử dụng như sau:

- Để truy cập RMAN:
    - Mở của sổ CMD.
    - Thực thi câu lệnh: RMAN




- Sau khi kết nối tới RMAN thành công, ta tiến hành kết nối tới target database và thực hiện các thao tác sao lưu và phục hồi trên target database. Cách thức thực hiện:
    - Chứng thực bằng mật khẩu:

```
CONNECT TARGET USERNAME/PASSWORD
```

Khi kết nối bằng câu lệnh này, RMAN sẽ mặc định kết nối tới CDB$ROOT của của Oracle. Tức là khi thực hiện backup và recovery ta sẽ thực hiện trên toàn bộ CDB.

- Để kết nối tới các PDB cụ thể, ta chỉ định thêm tên Net Service Name bằng cách sử dụng câu lệnh

```
CONNECT TARGET USERNAME/PASSWORD@SERVICENAME
```

- Trước tiên cần đảm bảo rằng LOG_MODE có giá trị là ARCHIVELOG bằng cách sử dụng câu lệnh:

```
SELECT LOG_MODE FROM V$DATABASE;
```

- Nếu không thực hiện các câu lệnh sau:

```
SHUTDOWN IMMEDIATE;
STARTUP MOUNT;
ALTER DATABASE ARCHIVELOG;
ALTER DATABASE OPEN;
```

- Thực hiện sao lưu toàn bộ cơ sở dữ liệu bằng lệnh:

```
BACKUP DATABASE
```

- Sau khi backup thành công ta có thể sử dụng câu lệnh sau để xem thông tin tóm tắt của các bản cập nhật:

```
LIST BACKUP OF DATABASE SUMMARY;
```

### 2.4.2. Recovery.

- Trước hết phải kết nối đến cơ sở dữ liệu đích.

```
 RESTORE DATABASE PREVIEW SUMMARY;
```

- Đảm bảo Mount CSDL bằng lệnh:

```
STARTUP FORCE MOUNT;
```

- Restore cơ sở dữ liệu:

```
RESTORE DATABASE;
```

- Recovery database:

```
RECOVER DATABASE;
```

- Open database sau khi restore:

```
ALTER DATABASE OPEN;
```

- Ngoài ra ta còn có thể backup/restore và recover các đơn vị cụ thể không nhất thiết phải là cả database như tablespace, datafile, control file,... với cú pháp tương ứng với từng loại.

## V. TÀI LIỆU THAM KHẢO


[1] Tài liệu lý thuyết môn học An toàn và bảo mật trong hệ thống thông tin.

[2] Tài liệu thực hành môn học An toàn và bảo mật trong hệ thống thông tin.

[3] Tài liệu lý thuyết môn học Chuyên Đề Hệ Quản Trị Cơ Sở Dữ Liệu Nâng Cao.