

**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: AN TOÀN VÀ BẢO MẬT DỮ LIỆU  
TRONG HỆ THỐNG THÔNG TIN**

**NHÓM THỰC HIỆN – 20H3T-01:**

**MSSV: 20120049 – HỌ TÊN: Nguyễn Hải Đăng**

**MSSV: 20120138 – HỌ TÊN: Lê Thành Nam**

**MSSV: 20120187 – HỌ TÊN: Nguyễn Viết Thái**

**MSSV: 20120289 – HỌ TÊN: Võ Minh Hiếu**

**Giảng viên lý thuyết: TS. Phạm Thị Bạch Huệ**

**Giảng viên thực hành: ThS. Lương Vĩ Minh – Tiết Gia Hồng**

**Lớp lý thuyết: 20\_1**

**Học kỳ - Niên khoá: HK2 - 2022-2023**

# MỤC LỤC

|   |           |
|---|-----------|
| <b>THÔNG TIN THÀNH VIÊN .....</b>   | <b>3</b>  |
| <b>BẢNG PHÂN CÔNG CÔNG VIỆC TRONG ĐỒ ÁN VÀ MỨC ĐỘ HOÀN THÀNH..3</b>   |           |
| <b>I. Phân hệ 1: Dành cho người quản trị cơ sở dữ liệu.....</b>   | <b>5</b>  |
| 1. Quản lý user/role.....   | 5         |
| 2. Quyền trên các đối tượng dữ liệu .....   | 6         |
| 3. Cấp role cho user/role .....   | 7         |
| 4. Xem danh sách table/view hiện có và tạo table.....   | 7         |
| 5. Quyền hệ thống.....  | 9         |
| <b>II. Phân hệ 2: Tạo và áp đặt chính sách bảo mật, mã hóa và ghi vết người dùng .....</b>  | <b>9</b>  |
| 1. Chính sách điều khiển truy cập (Access Control) .....  | 9         |
| 2. Mã hóa dữ liệu .....   | 15        |
| 3. Nhãn an toàn - Oracle Label Security .....   | 19        |
| <i>Câu a: Hãy gán nhãn cho 03 người dùng trong hệ thống. ....</i>   | <i>24</i> |
| <i>Câu b: Hãy cho biết cách thức phát tán dòng thông báo t1 đến tất cả trưởng phòng phụ trách tất cả các lĩnh vực không phân biệt chi nhánh. ....</i> | <i>25</i> |
| <i>Câu c: Hãy cho biết cách thức phát tán dòng thông báo t2 đến trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung.....</i>                        | <i>26</i> |
| <i>Câu d: Em hãy cho thêm một số kịch bản phát tán dữ liệu nữa trên mô hình OLS đã cài đặt.....</i>   | <i>27</i> |
| 4. Ghi vết hệ thống – Audit .....   | 28        |
| <i>Câu a: Những người đã cập nhật trường THOIGIAN trong quan hệ PHANCONG.....</i>   | <i>28</i> |
| <i>Câu b: Những người đã đọc trên trường LUONG và PHUCAP của người khác. ....</i>   | <i>29</i> |
| <i>Câu c: Một người không thuộc vai trò “Tài chính” nhưng đã cập nhật thành công trên trường LUONG và PHUCAP.....</i>                                 | <i>30</i> |
| <i>Câu d: Kiểm tra nhật ký hệ thống. ....</i>   | <i>31</i> |
| <b>III. Tài liệu tham khảo .....</b>  | <b>32</b> |

# **THÔNG TIN CHUNG VỀ ĐỒ ÁN**

## **THÔNG TIN THÀNH VIÊN**

**Bảng thông tin thành viên của nhóm 20H3T-01 trong đồ án:**

| <b>MÃ SỐ SINH VIÊN</b> | <b>HỌ VÀ TÊN</b> | <b>PHẦN TRĂM ĐÓNG GÓP</b> |
|------------------------|------------------|---------------------------|
| 20120049               | Nguyễn Hải Đăng  | 25%                       |
| 20120138               | Lê Thành Nam     | 25%                       |
| 20120187               | Nguyễn Viết Thái | 25%                       |
| 20120289               | Võ Minh Hiếu     | 25%                       |

## **BẢNG PHÂN CÔNG CÔNG VIỆC TRONG ĐỒ ÁN VÀ MỨC ĐỘ HOÀN THÀNH**

**Bảng phân công công việc của nhóm 20H3T-01 trong đồ án:**

| <b>Phân hệ</b> | <b>CÔNG VIỆC</b>  | <b>NGƯỜI THỰC HIỆN</b> | <b>MỨC ĐỘ HOÀN THÀNH</b> |
|----------------|---|------------------------|--------------------------|
| 1              | Tạo database + Insert dữ liệu.  | Lê Thành Nam           | 100% - Đã hoàn thành     |
| 1              | Xem danh sách người dùng trong hệ thống.  | Lê Thành Nam           | 100% - Đã hoàn thành     |
| 1              | Thông tin về quyền (privileges) của mỗi user/role trên các đối tượng dữ liệu.   | Nguyễn Viết Thái       | 100% - Đã hoàn thành     |
| 1              | Cho phép tạo mới, xóa, sửa (hiệu chỉnh) user hoặc role.   | Nguyễn Viết Thái       | 100% - Đã hoàn thành     |
| 1              | Cho phép thực hiện việc cấp quyền: cấp quyền cho user, cấp quyền cho role, cấp role cho user. Quá trình cấp quyền có tùy chọn là có cho phép người được cấp quyền có thể cấp quyền đó cho user/role khác hay không (có chỉ định WITH GRANT OPTION hay không). Quyền, select, update thì cho phép phân quyền | Nguyễn Hải Đăng        | 100% - Đã hoàn thành     |

|   |   |                  |                      |
|---|---|------------------|----------------------|
|   | tính đến mức cột; quyền insert, delete thì không.                                     |                  |                      |
| 1 | Cho phép thu hồi quyền từ người dùng/ role.   | Võ Minh Hiếu     | 100% - Đã hoàn thành |
| 1 | Cho phép kiểm tra quyền của các chủ thể vừa được cấp quyền..                          | Võ Minh Hiếu     | 100% - Đã hoàn thành |
| 1 | Cho phép chỉnh sửa quyền của user/ role.  | Nguyễn Hải Đăng  | 100% - Đã hoàn thành |
| 2 | Tạo database, Insert dữ liệu, Access Control: Chính sách 1.                           | Lê Thành Nam     | 100% - Đã hoàn thành |
| 2 | Access Control: Chính sách 2, 3.  | Nguyễn Viết Thái | 100% - Đã hoàn thành |
| 2 | Access Control: Chính sách 4, 5.  | Võ Minh Hiếu     | 100% - Đã hoàn thành |
| 2 | Access Control: Chính sách 6; Giải quyết xung đột chính sách.                         | Nguyễn Hải Đăng  | 100% - Đã hoàn thành |
| 2 | Mã hóa dữ liệu: Viết các hàm mã hóa, giải mã dữ liệu.                                 | Lê Thành Nam     | 100% - Đã hoàn thành |
| 2 | Mã hóa dữ liệu: Xây dựng chiến lược mã hóa, viết code mã hóa theo chiến lược đã chọn. | Võ Minh Hiếu     | 100% - Đã hoàn thành |
| 2 | OLS: Tạo chính sách OLS, gán nhãn người dùng và dữ liệu.                              | Nguyễn Hải Đăng  | 100% - Đã hoàn thành |
| 2 | Audit: Ghi vết hệ thống.  | Nguyễn Viết Thái | 100% - Đã hoàn thành |

# PHẦN BÁO CÁO

## I. Phân hệ 1: Dành cho người quản trị cơ sở dữ liệu

### 1. Quản lý user/role

QUẢN TRỊ VIÊN

User & Role Privilege Role Table & View System Privilege Audit

XIN CHÀO QLDA! ĐĂNG XUẤT

**USERS** SELECT

|   | USERNAME         | USER_ID    | CREATED          |
|---|------------------|------------|------------------|
| ▶ | SYS              | 0          | 28/09/2021 04:32 |
|   | AUDSYS           | 8          | 28/09/2021 04:32 |
|   | SYSTEM           | 9          | 28/09/2021 04:32 |
|   | SYSDG            | 2147483617 | 28/09/2021 04:32 |
|   | SYSDG            | 2147483618 | 28/09/2021 04:32 |
|   | SYSKM            | 2147483619 | 28/09/2021 04:32 |
|   | SYSRAC           | 2147483620 | 28/09/2021 04:32 |
|   | OUTLN            | 13         | 28/09/2021 04:32 |
|   | CSADMIN_INTERNAL | 24         | 28/09/2021 04:32 |

Create Users  
Delete User  
Update User

Tim kiem User

**ROLES**

|   | ROLE                 | ROLE_ID | PASSWORD_REQUIRED |
|---|----------------------|---------|-------------------|
| ▶ | CONNECT              | 2       | NO                |
|   | RESOURCE             | 3       | NO                |
|   | DBA                  | 4       | NO                |
|   | PDB_DBA              | 5       | NO                |
|   | AUDIT_ADMIN          | 6       | NO                |
|   | AUDIT_VIEWER         | 7       | NO                |
|   | SELECT_CATALOG_ROLE  | 10      | NO                |
|   | EXECUTE_CATALOG_ROLE | 11      | NO                |
|   | CAPTURE_ADMIN        | 12      | NO                |

Create Role  
Delete Role  
Update Role

Tim kiem Role

ĐÓNG

Trong ứng dụng này, chúng em tạo user admin là QLDA với các role và quyền như bên dưới:

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

Sử dụng các câu lệnh sau để thực hiện chức năng:

- Xem danh sách user hiện có:
  - SELECT USERNAME, USER\_ID, CREATED FROM ALL\_USERS
- Xem danh sách role hiện có:
  - SELECT ROLE, ROLE\_ID, PASSWORD\_REQUIRED FROM DBA\_ROLES
- Tạo user:
  - CREATE USER username IDENTIFIED BY password;
- Cập nhật user (mật khẩu)

- ALTER USER username IDENTIFIED BY new\_password;
- Xóa user:
  - DROP USER username;
- Tạo role (có hoặc không có mật khẩu):
  - CREATE ROLE username {IDENTIFIED BY password};
- Cập nhật role (mật khẩu):
  - ALTER ROLE username {NOT IDENTIFIED/IDENTIFIED BY new\_password};
- Xóa role:
  - DROP ROLE rolename;

## 2. Quyền trên các đối tượng dữ liệu

QUẢN TRỊ VIÊN

User & Role Privilege Role Table & View System Privilege Audit

XIN CHÀO QLDA! ĐĂNG XUẤT

CẤP QUYỀN CHO ROLE/USER HỖY QUYỀN CHO ROLE/USER

Tim kiem Grantee ROLETEST Search SELECT

**TABLE**

|   | GRANTEE | OWNER | TABLE_NAME   | GRANTOR | PRIVILEGE | GRANTABLE | HIERARCHY | COMMON | TYPE  | INHERITED |
|---|---------|-------|--------------|---------|-----------|-----------|-----------|--------|-------|-----------|
| ▶ | NV      | QLDA  | QLDA_PHON... | QLDA    | SELECT    | NO        | NO        | NO     | TABLE | NO        |
|   | NS      | QLDA  | QLDA_PHON... | QLDA    | SELECT    | NO        | NO        | NO     | TABLE | NO        |
|   | NS      | QLDA  | QLDA_PHON... | QLDA    | INSERT    | NO        | NO        | NO     | TABLE | NO        |
|   | NS      | QLDA  | QLDA_PHON... | QLDA    | UPDATE    | NO        | NO        | NO     | TABLE | NO        |
|   | QL      | QLDA  | QLDA_PHON... | QLDA    | SELECT    | NO        | NO        | NO     | TABLE | NO        |
|   | TP      | QLDA  | QLDA_PHON... | QLDA    | SELECT    | NO        | NO        | NO     | TABLE | NO        |
|   | TC      | QLDA  | QLDA_PHON... | QLDA    | SELECT    | NO        | NO        | NO     | TABLE | NO        |
|   | TA      | QLDA  | QLDA_PHON... | QLDA    | SELECT    | NO        | NO        | NO     | TABLE | NO        |

**COLUMN**

|   | GRANTEE | OWNER | TABLE_NAME    | COLUMN_NAME | GRANTOR | PRIVILEGE | GRANTABLE | COMMON | INHE |
|---|---------|-------|---------------|-------------|---------|-----------|-----------|--------|------|
| ▶ | QL      | QLDA  | QLDA_NHANVIEN | NGAYSINH    | QLDA    | UPDATE    | NO        | NO     | NO   |
|   | QL      | QLDA  | QLDA_NHANVIEN | DIACHI      | QLDA    | UPDATE    | NO        | NO     | NO   |
|   | QL      | QLDA  | QLDA_NHANVIEN | SODT        | QLDA    | UPDATE    | NO        | NO     | NO   |
|   | TP      | QLDA  | QLDA_NHANVIEN | NGAYSINH    | QLDA    | UPDATE    | NO        | NO     | NO   |
|   | TP      | QLDA  | QLDA_NHANVIEN | DIACHI      | QLDA    | UPDATE    | NO        | NO     | NO   |
|   | TP      | QLDA  | QLDA_NHANVIEN | SODT        | QLDA    | UPDATE    | NO        | NO     | NO   |

ĐÓNG

- Xem thông tin quyền trên TABLE và COLUMN của các user và role trên hệ thống:
  - select \* from DBA\_TAB\_PRIVS where TABLE\_NAME LIKE 'QLDA\_%' OR TABLE\_NAME LIKE 'V\_QLDA\_%'
  - select \* from DBA\_COL\_PRIVS where TABLE\_NAME LIKE 'QLDA\_%'
- Cấp quyền cho role/user:
  - Đối với quyền SELECT: trên thực tế, Oracle không cho phép cấp quyền SELECT trên thuộc tính trực tiếp trên một table, nên khi cấp quyền SELECT trên thuộc tính, chúng em sẽ tạo một view với table và các thuộc tính được truy xuất và cấp quyền trên view đó.
  - Đối với quyền UPDATE: Oracle cho phép cấp quyền đến mức thuộc tính.

- Quyền INSERT, UPDATE: chỉ cấp quyền cho toàn bộ table, không cấp riêng cho thuộc tính.
- Câu lệnh cấp quyền cho role/user
  - GRANT privilege TO username/rolename
- Lấy lại quyền của user/role:
  - REVOKE privilege TO username/rolename

### 3. Cấp role cho user/role

QUẢN TRỊ VIÊN

User & Role Privilege Role Table & View System Privilege Audit

XIN CHÀO QLDA! ĐĂNG XUẤT

CẤP ROLE CHO USER/ROLE HỖY ROLE CỦA USER/ROLE

Tìm kiếm User Tìm kiếm Role

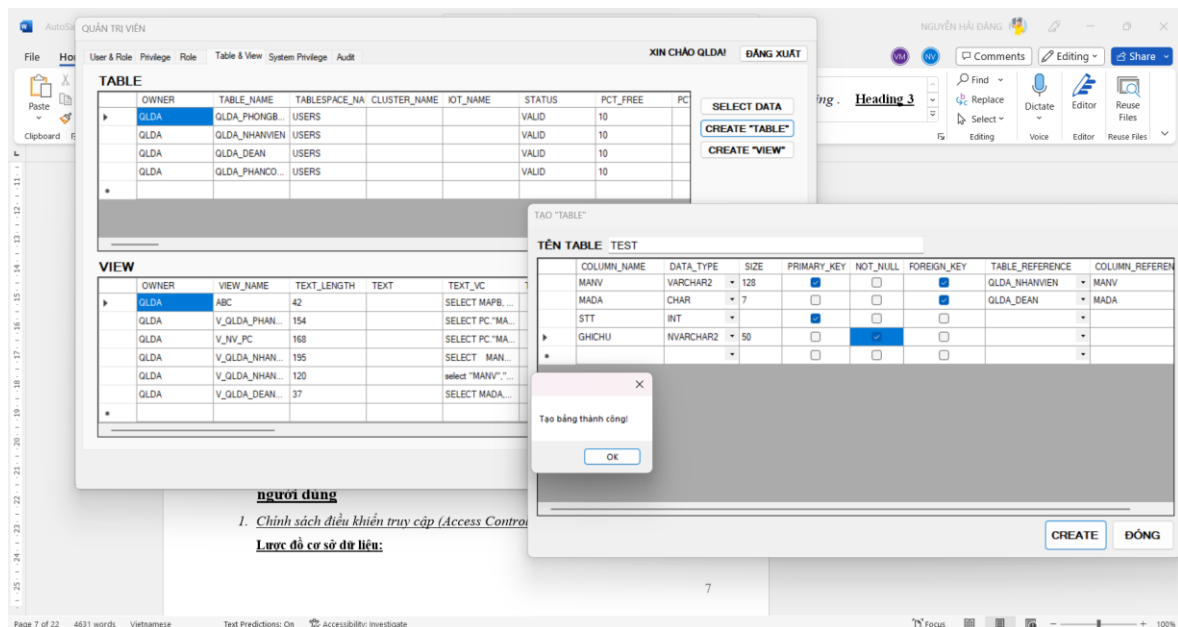
SELECT

|   | GRANTEE         | GRANTED_ROLE    | ADMIN_OPTION | DELEGATE_OPTK | DEFAULT_ROLE | COMMON | INHERITED |
|---|-----------------|-----------------|--------------|---------------|--------------|--------|-----------|
| ▶ | GSMROOTUSER     | GSMROOTUSE...   | NO           | NO            | YES          | NO     | NO        |
|   | QLDA            | DBA             | YES          | NO            | YES          | NO     | NO        |
|   | NV              | CONNECT         | NO           | NO            | YES          | NO     | NO        |
|   | SYS             | AUDIT_VIEWER    | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | CAPTURE_ADMIN   | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | GATHER_SYST...  | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | OPTIMIZER_PR... | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | EM_EXPRESS...   | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | GSMADMIN_RO...  | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | XDB_WEBSERV...  | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | SODA_APP        | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | DATAPATCH_R...  | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | JAVAUERPRIV     | YES          | NO            | YES          | YES    | NO        |
|   | SYS             | ORDADMIN        | YES          | NO            | YES          | YES    | NO        |
|   | DBA             | EXECUTE_CAT...  | NO           | NO            | YES          | YES    | NO        |
|   | SYSTEM          | DBA             | NO           | NO            | YES          | YES    | NO        |
|   | SELECT_CATAL... | HS_ADMIN_SEL... | NO           | NO            | YES          | YES    | NO        |

ĐÓNG

- Oracle cho phép cấp Role cho một user và cả cấp Role cho Role.
- Câu lệnh thực hiện cấp Role cho User/Role:
  - GRANT role TO username/rolename;
- Việc cấp role cho role là cần thiết vì có thể ta cần phải cấp role hệ thống cho role (như role DBA, CONNECT)

### 4. Xem danh sách table/view hiện có và tạo table

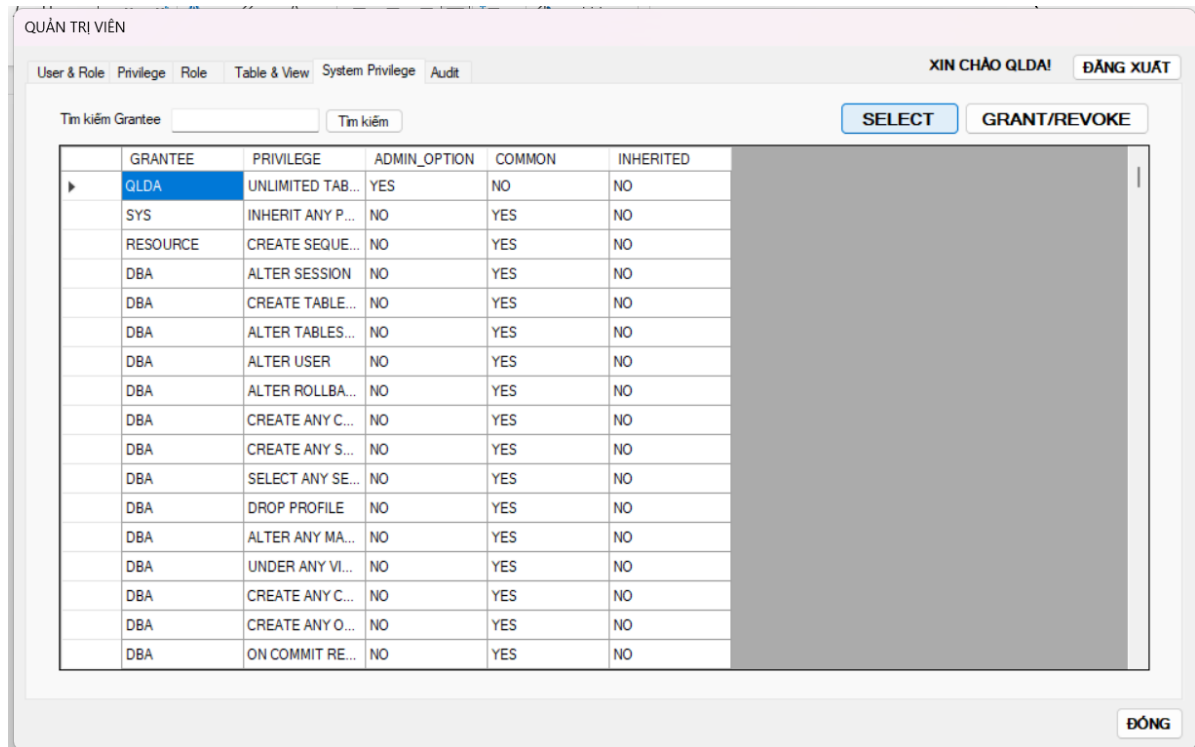


- Xem danh sách table hiện có của một owner đang đăng nhập:
  - o `select * from dba_tables where owner = :owner`
  - o Với owner là user đang đăng nhập.
- Xem danh sách view hiện có của một owner đang đăng nhập:
  - o `select * from dba_views where owner = :owner`
  - o Với owner là user đang đăng nhập.
- Code tạo table:

```
private void createButton_Click(object sender, EventArgs e)
{
    try
    {
        if (tabelTextBox.TextLength == 0)
        {
            MessageBox.Show("Vui lòng nhập tên bảng!");
            return;
        }
        if (attributeTable.Rows.Count == 0)
        {
            MessageBox.Show("Vui lòng nhập thuộc tính cho bảng!");
            return;
        }
        string tableName = tabelTextBox.Text;
        string sql = "CREATE TABLE " + tableName + "(";
        foreach (DataGridViewRow row in attributeTable.Rows)
        {
            if (row.Index == attributeTable.Rows.Count - 1)
            {
                break;
            }
            if (row.Cells["COLUMN_NAME"].Value == null || row.Cells["COLUMN_NAME"].Value.ToString() == "")
            {
                MessageBox.Show("Vui lòng nhập đầy đủ thuộc tính!");
                return;
            }
            sql += row.Cells["COLUMN_NAME"].Value.ToString() + " " + row.Cells["DATA_TYPE"].Value.ToString() + " ";
            if (row.Cells["SIZE"].Value != null) sql += "(" + row.Cells["SIZE"].Value.ToString() + ") ";
            sql += Convert.ToBoolean(row.Cells["NOT_NULL"].Value) ? " NOT NULL " : "";
            if (row.Cells["FOREIGN_KEY"].Value != null && Convert.ToBoolean(row.Cells["FOREIGN_KEY"].Value))
            {
                sql += " REFERENCES " + row.Cells["TABLE_REFERENCE"].Value.ToString() + "(" + row.Cells["COLUMN_REFERENCE"].Value.ToString() + ") ";
            }
            sql += ",";
        }
        string PK = "";
        foreach (DataGridViewRow row in attributeTable.Rows)
        {
            if (row.Cells["PRIMARY_KEY"].Value != null && Convert.ToBoolean(row.Cells["PRIMARY_KEY"].Value))
            {
                if (PK == "") PK += "PRIMARY KEY(";
                PK += row.Cells["COLUMN_NAME"].Value.ToString() + ",";
            }
        }
        if (PK != "")
        {
            PK = PK.Remove(PK.Length - 1, 1);
            PK += ")";
        }
        sql += PK;
        sql = sql.Remove(sql.Length - 1, 1);
        sql += ")";
        OracleCommand cmd = new OracleCommand(sql, LoginUI.con);
        cmd.CommandType = CommandType.Text;
        cmd.ExecuteNonQuery();
        MessageBox.Show("Tạo bảng thành công!");
    }
    catch (OracleException ex)
    {
        MessageBox.Show(ex.Message);
    }
}
```



## 5. Quyền hệ thống



- Ngoài quyền đối tượng, Oracle còn một loại quyền là quyền hệ thống và chúng ta có thể quyền hệ thống cho các chủ thể.
- Câu lệnh cấp quyền hệ thống cho role/user
  - o GRANT privilege TO username/rolename
- Lấy lại cấp quyền hệ thống của user/role:
  - o REVOKE privilege TO username/rolename

## II. Phân hệ 2: Tạo và áp đặt chính sách bảo mật, mã hóa và ghi vết người dùng

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

### Lược đồ cơ sở dữ liệu:

NHANVIEN (MANV, TENNV, PHAI, NGAYSINH, DIACHI, SODT, LUONG, PHUCAP, VAITRO, MANQL, PHG)

PHONGBAN (MAPB, TENPB, TRPHG)

DEAN (MADA, TENDA, NGAYBD, PHONG)

PHANCONG (MANV, MADA, THOIGIAN)

THONGBAO (MATB, NOIDUNG, DIADIEM, OLS\_THONGBAO)

### Phát biểu lại các chính sách:

- Chính sách 1: Những người có VAITRO là “Nhân viên” cho biết đó là một nhân viên bình thường, không kiêm những công việc nào khác. Những người dùng có VAITRO là “Nhân viên” có các quyền được mô tả như sau:
  - Có quyền xem tất cả các thuộc tính trên quan hệ NHANVIEN và PHANCONG liên quan đến nhân viên đó.
  - Có thể sửa trên các thuộc tính NGAYSINH, DIACHI, SODT liên quan đến chính nhân viên đó.
  - Có thể xem dữ liệu của toàn bộ quan hệ PHONGBAN và DEAN.
  - Hiện tại có 300 nhân viên trong toàn hệ thống.
- Chính sách 2: Những người dùng có VAITRO là “QL trực tiếp” nếu họ phụ trách quản lý trực tiếp nhân viên khác. Nhân viên Q là quản lý trực tiếp của nhân viên N, có quyền được mô tả như sau:
  - Q có quyền như là một nhân viên thông thường (vai trò “Nhân viên”). Ngoài ra, với các dòng dữ liệu trong quan hệ NHANVIEN liên quan đến các nhân viên N mà Q quản lý trực tiếp thì Q được xem tất cả các thuộc tính, trừ thuộc tính LUONG và PHUCAP.
  - Có thể xem các dòng trong quan hệ PHANCONG liên quan đến chính Q và các nhân viên N được quản lý trực tiếp bởi Q.
  - Hệ thống S hiện tại có 20 người là quản lý trực tiếp.
- Chính sách 3: Những người dùng có VAITRO là “Trưởng phòng” cho biết đó là một nhân viên kiêm nhiệm thêm vai trò trưởng phòng. Một người dùng T có VAITRO là “Trưởng phòng” có quyền được mô tả như sau:
  - T có quyền như là một nhân viên thông thường (vai trò “Nhân viên”). Ngoài ra, với các dòng trong quan hệ NHANVIEN liên quan đến các nhân viên

- thuộc phòng ban mà T làm trưởng phòng thì T có quyền xem tất cả các thuộc tính, trừ thuộc tính LUONG và PHUCAP.
- Có thể thêm, xóa, cập nhật, **xem** trên quan hệ PHANCONG liên quan đến các nhân viên thuộc phòng ban mà T làm trưởng phòng.
  - Hệ thống S hiện tại có 8 người là trưởng phòng.
  - **Chính sách 4:** Những người có VAITRO là “**Tài chính**” cho biết đó là một nhân viên phụ trách công việc tài chính tiền lương của công ty. Một người dùng TC có vai trò “Tài chính” có quyền được mô tả như sau:
    - TC có quyền như là một nhân viên thông thường (vai trò “Nhân viên”).
    - TC có quyền xem toàn bộ quan hệ NHANVIEN, có thể chỉnh sửa trên thuộc tính LUONG và PHUCAP (thừa hành ban giám đốc) nhưng **chỉ có thể chỉnh sửa thuộc tính NGAYSINH, DIACHI, SODT của chính mình.**
    - TC có quyền xem toàn bộ quan hệ PHANCONG.
    - Hệ thống S hiện tại có 5 người phụ trách công tác tài chính.
  - **Chính sách 5:** Những người có VAITRO là “**Nhân sự**” cho biết đó là một nhân viên phụ trách công tác nhân sự trong công ty. Một người dùng NS có vai trò “Nhân sự” có quyền được mô tả như sau:
    - NS có quyền như là một nhân viên thông thường (vai trò “Nhân viên”).
    - Được quyền **xem**, thêm, cập nhật trên quan hệ PHONGBAN.
    - Thêm, cập nhật dữ liệu trong quan hệ NHANVIEN với giá trị các trường LUONG, PHUCAP là mang giá trị mặc định là NULL, không được xem LUONG, PHUCAP của người khác và không được cập nhật trên các trường LUONG, PHUCAP. Nhưng **được chỉnh sửa thuộc tính NGAYSINH, DIACHI, SODT của chính mình.**
    - Hệ thống S hiện tại có 5 người phụ trách công tác nhân sự.
  - **Chính sách 6:** Những người dùng có VAITRO là “Trưởng đề án” cho biết đó là nhân viên là trưởng các đề án. Một người dùng là “Trưởng đề án” có quyền được mô tả như sau:
    - Có quyền như là một nhân viên thông thường (vai trò “Nhân viên”).
    - Được quyền thêm, xóa, cập nhật, **xem** trên quan hệ DEAN.
    - Hệ thống S hiện tại có 3 người phụ trách công tác trưởng đề án.

#### **Kịch bản cài đặt:**

- Đối với bảng NHANVIEN: tạo 2 view:
  - V\_QLDA\_NHANVIEN sẽ SELECT toàn bộ table của quan hệ NHANVIEN, cấp quyền trên view này cho **nhân viên, trưởng đề án, tài chính và giám đốc**, sau đó cài VPD trên view tương ứng với các quyền mà mỗi vai trò được làm.
  - V\_QLDA\_NHANVIEN\_NS sẽ SELECT toàn bộ table của quan hệ NHANVIEN, cấp quyền trên view này cho **quản lý, trưởng phòng và**

**nhân sự.** View này có đặc điểm là người dùng chỉ được xem thuộc LUONG, PHUCAP của chính mình, các dòng dữ liệu còn lại sẽ bị DECODE thành NULL trên 2 thuộc tính đó (content-based access control). Sau đó cài VPD trên view tương ứng với các quyền mà mỗi vai trò được làm.

- Đối với bảng PHANCONG: tạo 1 view:
  - o V\_QLDA\_PHANCONG\_QL sẽ SELECT toàn bộ table của quan hệ NHANVIEN nhưng chỉ trả về những dòng dữ liệu liên quan đến nhân viên mà một người quản lý đó đang quản lý, **cấp quyền trên view này cho quản lý.**
  - o **Các vai trò khác SELECT trên table của quan hệ.**
- Đối với bảng PHONGBAN và DEAN: các vai trò được cấp quyền trực tiếp trên table của 2 quan hệ PHONGBAN và DEAN
- Chính sách 1:
  - o Chủ thể: những người dùng có role “NV” – Nhân viên.
  - o Cơ chế sử dụng: RBAC, VPD.
  - o Quyền:
    - NHANVIEN: SELECT, UPDATE(NGAYSINH, SĐT, DIACHI)
    - PHANCONG: SELECT
    - PHONGBAN: SELECT
    - DEAN: SELECT
  - o Kịch bản cài đặt:
    - Cấp quyền SELECT, UPDATE(NGAYSINH, SĐT, DIACHI) trên view V\_QLDA\_NHANVIEN và quyền SELECT trên bảng PHANCONG cho role NV, sau đó dùng VPD để giới hạn quyền truy cập để role NV chỉ truy cập được dòng của bản thân.
    - Cấp quyền SELECT của bảng PHONGBAN, DEAN cho role NV.
- Chính sách 2:
  - o Chủ thể: những người dùng có role là “QL” - Quản lý trực tiếp.
  - o Quyền:
    - NHANVIEN: SELECT, UPDATE(NGAYSINH, SODT, DIACHI) của bản thân.
    - NHANVIEN: SELECT (-LUONG, -PHUCAP) nhân viên do mình quản lý.
    - PHANCONG: SELECT.
    - PHONGBAN: SELECT.
    - DEAN: SELECT.
  - o Kịch bản:
    - Sử dụng view V\_QLDA\_NHANVIEN\_NS để QL chỉ có thể thấy được LUONG và PHUCAP của mình. Cấp quyền SELECT, UPDATE(NGAYSINH, SODT, DIACHI) trên view đó. Sau đó, cài

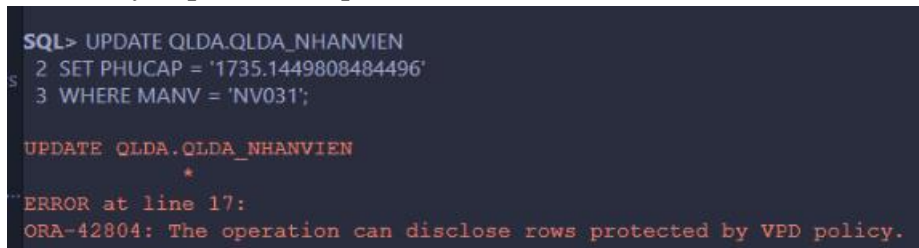
- đặt VPD trên View này với hàm chính sách để Quản lý chỉ được SELECT thông tin của mình và SELECT(- LUONG, -PHUCAP) của nhân viên do mình quản lý. Đồng thời cài VPD trên view V\_QLDA\_NHANVIEN\_NS cho phép quản lý UPDATE(NGAYSINH, SODT, DIACHI) của mình.
- Sử dụng view V\_QLDA\_PHANCONG\_QL để có thể lấy được những dòng liên quan đến quản lý và những nhân viên được quản lý từ quan hệ PHANCONG. Sau đó cho người dùng (Quản lý) quyền SELECT view này.
    - Cơ chế sử dụng: VPD, RBAC, CBAC (Content-Based Access Control).
  - **Chính sách 3:**
    - Chủ thể: những người dùng có role là “TP” – Trưởng phòng.
    - Quyền:
      - NHANVIEN: SELECT, UPDATE(NGAYSINH, DIACHI, SODT) của chính mình.
      - NHANVIEN: SELECT (-LUONG, -PHUCAP) của nhân viên thuộc phòng ban của mình.
      - PHANCONG: SELECT, INSERT, DELETE, UPDATE
      - PHONGBAN: SELECT.
      - DEAN: SELECT.
    - Kịch bản:
      - Cấp quyền SELECT, UPDATE(NGAYSINH, DIACHI, SODT) trên View V\_QLDA\_NHANVIEN\_NS. Sau đó, cài đặt VPD trên view này với hàm chính sách là trưởng phòng chỉ được SELECT(-LUONG, -PHUCAP) các nhân viên thuộc phòng ban của mình. Đồng thời, cài VPD trên View này với hàm chính sách để Trưởng phòng có thể UPDATE(NGAYSINH, DIACHI, SODT) của chính mình.
      - Cài đặt VPD trên table QLDA\_PHANCONG với hàm chính sách cho phép Trưởng phòng chỉ được SELECT, INSERT, DELETE, UPDATE đối với những dòng nhân viên thuộc phòng ban của Trưởng phòng đó.
    - Cơ chế sử dụng: RBAC, VPD.
  - **Chính sách 4:**
    - Chủ thể: người dùng có role là ‘TC’ – Tài chính.
    - Quyền:
      - NHANVIEN: SELECT, UPDATE (LUONG, PHUCAP)
      - NHANVIEN: UPDATE (NGAYSINH, DIACHI, SODT) của chính mình
      - PHANCONG: SELECT
      - PHONGBAN: SELECT

- DEAN: SELECT
- Cơ chế sử dụng: RBAC, VPD.
- Kịch bản cài đặt:
  - Cấp quyền SELECT, UPDATE (LUONG, PHUCAP, NGAYSINH, DIACHI, SODT) trên view V\_QLDA\_NHANVIEN. Sử dụng VPD trên view này để cài đặt chính sách các nhân viên tài chính chỉ có thể chỉnh sửa (NGAYSINH, DIACHI, SODT) của chính mình.
  - Sử dụng RBAC để cấp quyền SELECT trên bảng QLDA\_PHANCONG.
- Chính sách 5:
  - Chủ thể: những người dùng có vai trò là “NS” – Nhân sự.
  - Quyền:
    - PHONGBAN: SELECT, INSERT, UPDATE
    - NHANVIEN: SELECT, INSERT (LUONG = NULL, PHU CAP = NULL), UPDATE ( -LUONG, - PHUCAP)
    - PHANCONG: SELECT
    - DEAN: SELECT
  - Cơ chế sử dụng: RBAC, CBAC (CONTENT BASE ACCESS CONTROL)
  - Kịch bản cài đặt:
    - Sử dụng RBAC để cấp quyền SELECT, INSERT, UPDATE trên quan hệ PHONGBAN.
    - Sử dụng CBAC tạo view V\_QLDA\_NHANVIEN\_NS chứa các thuộc tính của quan hệ NHANVIEN và dùng hàm DECODE để chỉ có thể thấy LUONG, PHUCAP của mình. Sau đó sử dụng RBAC cấp quyền INSERT, UPDATE trên view V\_QLDA\_NHANVIEN\_NS ( không chứa thuộc tính LUONG và PHUCAP).
- Chính sách 6:
  - Chủ thể: những người dùng có role là “TA” – Trưởng đề án.
  - Quyền:
    - NHANVIEN: SELECT, UPDATE (NGAYSINH, SĐT, DIACHI).
    - PHONGBAN: SELECT.
    - DEAN: SELECT, INSERT, DELETE, UPDATE.
    - PHANCONG: SELECT.
  - Kịch bản:
    - Sử dụng View V\_QLDA\_NHANVIEN có thể xem toàn bộ quan hệ NHANVIEN, cấp quyền SELECT, UPDATE trên view đó. Sau đó, sử dụng VPD trên View này với hàm chính sách là trưởng đề án chỉ được quyền SELECT, UPDATE (NGAYSINH, SĐT, DIACHI) những dòng thông tin có liên quan đến chính mình.

- Trên quan hệ PHANCONG, sử dụng VPD trên table này với hàm chính sách là trường đề án chỉ được SELECT dòng dữ liệu phân công liên quan đến chính mình.
- Các quan hệ còn lại cho phép cấp quyền trên table của quan hệ, không dùng chính sách VPD.
- Sử dụng cơ chế RBAC, VPD.

## 2. Mã hóa dữ liệu

- Khái niệm: Mã hóa là quá trình biến đổi dữ liệu từ dạng văn bản bình thường sang dạng mã (không có nghĩa). Từ dạng mã muốn chuyển về cần phải giải mã.
- Ngữ cảnh: Chỉ có người dùng được cấp quyền mới xem được cột LUONG, PHUCAP trong bảng NHANVIEN.
- User thực hiện vai trò mã hóa: user admin QLDA
  - Chọn user admin để thực hiện mã hóa nhằm tránh các xung đột với các trigger được cài đặt bởi user admin khi thực hiện lệnh insert, update trên các thuộc tính được mã hóa.
- Mức mã hóa dữ liệu: Mức cơ sở dữ liệu
  - Lựa chọn mã hóa ở mức cơ sở dữ liệu vì nó cung cấp nhiều cấp độ mã hóa (cấp độ cột/thuộc tính, cấp độ bộ/dòng, cấp độ trang/khối).
  - Trong dự án lần này, chúng ta cần mã hóa trên hai thuộc tính là LUONG và PHUCAP, nên việc lựa chọn mã hóa ở mức cơ sở dữ liệu sẽ chỉ mã hóa những thông tin cần thiết, giúp mã hóa nhanh hơn, giảm dung lượng lưu trữ so với các mức mã hóa khác.
  - Mã hóa mức cơ sở dữ liệu cũng chống được các kiểu tấn công như: đánh cắp thiết bị lưu trữ, tấn công mức cơ sở dữ liệu (SQL Injection), người quản trị truy cập dữ liệu bất hợp pháp, ...
- Việc mã hóa không ảnh hưởng đến cấu trúc dữ liệu nhưng chúng ta không thể sử dụng chính sách VPD để điều khiển truy cập trên các thuộc tính được mã hóa – không thể set DBMS\_RLS.ALL\_ROWS của tham số sec\_relevant\_cols\_opt trong chính sách VPD (Lỗi: Quá trình có thể làm lộ dữ liệu các dòng được bảo vệ bởi chính sách VPD). Thay vào đó, chúng ta sẽ sử dụng chính sách CBAC để điều khiển truy cập trên các quan hệ chứa thuộc tính được mã hóa.



```
SQL> UPDATE QLDA.QLDA_NHANVIEN
2 SET PHUCAP = '1735.1449808484496'
3 WHERE MANV = 'NV031';

UPDATE QLDA.QLDA_NHANVIEN
*
ERROR at line 17:
ORA-42804: The operation can disclose rows protected by VPD policy.
```

- Thiết lập khóa: sử dụng chính sách tính toán ra khóa

- Không cần phải lưu khóa trong cơ sở dữ liệu vì có phải hạn chế truy cập tới bảng lưu khóa, duy trì việc giám sát truy cập vào bảng và rủi ro đối với việc can thiệp và thay đổi khóa bởi DBA.
  - Không sử dụng quản lý khóa bởi ứng dụng nhằm tránh mất khóa khi ứng dụng gặp sự cố.
  - Mỗi nhân viên sẽ có một khóa khác nhau nên việc tìm ra tất cả các khóa mà không có công thức sẽ không dễ dàng.
- Công thức tạo ra khóa:

```

FUNCTION CREATE_KEY(
    ma NVARCHAR2,
    seq NUMBER
) RETURN RAW DETERMINISTIC IS
    MS_RAW RAW(128);
    C_KEY RAW(128);
    T_RAW RAW(128);
BEGIN
    T_RAW := UTL_RAW.CAST_TO_RAW('ATBM_HTTT_T1' || seq);
    MS_RAW := UTL_RAW.CAST_TO_RAW(seq || ma);
    C_KEY := UTL_RAW.BIT_XOR(T_RAW, MS_RAW);
    RETURN C_KEY;
END CREATE_KEY;

```

- Gồm 2 tham số đầu vào là MANV và seq (số thứ tự lần thay đổi khóa).
  - Đầu tiên nối chuỗi 'ATBM\_HTTT\_T1' và seq rồi chuyển sang kiểu RAW.
  - Tiếp theo nối chuỗi seq và MANV rồi chuyển sang kiểu RAW.
  - Cuối cùng, KEY sẽ được tạo ra bằng cách BIT\_XOR hai kết quả trên.
- Lưu trữ khóa:
- Vì khóa được tính toán ra nên chúng ta chỉ cần lưu trữ công thức tạo ra khóa bằng cách xóa nó khỏi database và lưu vào thiết bị ngoại vi. Thiết bị này được cất giữ bởi người có thẩm quyền.
  - Các lưu trữ này sẽ tránh được cách rủi ro nếu bị đánh cắp cơ sở dữ liệu nhưng không có quyền giải mã.
- Phân phối khóa:
- Chỉ cấp quyền giải mã cho một số ROLE nhất định.
  - Cấp quyền xem dữ liệu thực thông qua các view, không cấp quyền insert, update trên các view đó.
  - Các trigger tự động mã hóa dữ liệu trên các thuộc tính mã hóa vừa được insert, update.
- Phục hồi khóa khi người dùng quên khóa:
- Vì khóa được tính toán ra nên không xảy ra việc người dùng quên khóa, user admin chỉ cần sử dụng các PROCEDURE để mã hóa và giải mã, tương tự cho người dùng có quyền.



- Công thức khóa được lưu sang thiết bị ngoại vi nên chỉ cần đảm bảo bảo quản tốt thiết bị ngoại vi thì sẽ không sợ mất khóa.
- Thay đổi khóa đồng loạt sau một thời gian:
  - Trong công thức tạo ra khóa có tham số là seq, nó là một tham số public, khi cần thay đổi khóa, admin chỉ cần thay đổi tham số seq sẽ đồng loạt tạo ra khóa mới cho tất cả nhân viên.
  - Các bước thay đổi khóa:
    - B1: Xóa trigger **auto\_encrypted\_nhanvien** (trigger này sẽ tự động mã hóa dữ liệu khi insert hay update).
    - B2: Giải mã dữ liệu với khóa hiện tại.
    - B3: Mã hóa dữ liệu với khóa mới (thay đổi tham số seq).
    - B4: Bật lại trigger **auto\_encrypted\_nhanvien**.
- Cách thực hiện mã hóa:
  - Tạo package Encrypt\_Decrypt gồm 2 function thực hiện mã hoá và giải mã dữ liệu truyền vào.
  - Tạo trigger tự động mã hoá khi thêm NHANVIEN hoặc cập nhật LUONG, PHUCAP.
  - Gán quyền thực thi package cho những đối tượng được phép xem thông tin LUONG, PHUCAP của NHANVIEN.

```

----- MÃ HOÁ -----
-- Mã hoá thông tin LUONG và PHUCAP
-- Tạo package hỗ trợ mã hoá - giải mã
CREATE OR REPLACE PACKAGE QLDA.ENCRYPT_DECRYPT
AS
    FUNCTION ENCRYPT_NHANVIEN_LUONG(
        P_IN IN NVARCHAR2,
        P_KEY IN CHAR
    ) RETURN RAW DETERMINISTIC;
    FUNCTION DECRYPT_NHANVIEN_LUONG(
        P_IN IN RAW,
        P_KEY IN CHAR
    ) RETURN NVARCHAR2 DETERMINISTIC;
    FUNCTION ENCRYPT_NHANVIEN_PHUCAP(
        P_IN IN NVARCHAR2,
        P_KEY IN CHAR
    ) RETURN RAW DETERMINISTIC;
    FUNCTION DECRYPT_NHANVIEN_PHUCAP(
        P_IN IN RAW,
        P_KEY IN CHAR
    ) RETURN NVARCHAR2 DETERMINISTIC;
    FUNCTION CREATE_KEY(
        ma NVARCHAR2,
        seq NUMBER
    ) RETURN RAW DETERMINISTIC;
    FUNCTION SEQ_NUM RETURN NUMBER DETERMINISTIC;
END ENCRYPT_DECRYPT;
/

```

```

-- Cài đặt các function trong package trên
CREATE OR REPLACE PACKAGE BODY QLDA.ENCRYPT_DECRYPT IS
    ENCRYPTION_TYPE PLS_INTEGER := DBMS_CRYPTO.ENCRYPT_DES +DBMS_CRYPTO.CHAIN_CBC +DBMS_CRYPTO.PAD_PKCS5;
    SEQ_NUMBER NUMBER := 111;
    FUNCTION ENCRYPT_NHANVIEN_LUONG(
        P_IN IN NVARCHAR2,
        P_KEY IN CHAR
    ) RETURN RAW DETERMINISTIC IS
        ENCRYPTED_RAW RAW(2000);
    BEGIN
        ENCRYPTED_RAW := DBMS_CRYPTO.ENCRYPT( SRC => UTL_RAW.CAST_TO_RAW(P_IN), TYP => ENCRYPTION_TYPE, KEY => UTL_RAW.CAST_TO_RAW(P_KEY) );
        RETURN ENCRYPTED_RAW;
    END ENCRYPT_NHANVIEN_LUONG;
    FUNCTION DECRYPT_NHANVIEN_LUONG(
        P_IN IN RAW,
        P_KEY IN CHAR
    ) RETURN NVARCHAR2 DETERMINISTIC IS
        DECRYPTED_RAW RAW(2000);
    BEGIN
        DECRYPTED_RAW := DBMS_CRYPTO.DECRYPT( SRC => P_IN, TYP => ENCRYPTION_TYPE, KEY => UTL_RAW.CAST_TO_RAW(P_KEY) );
        RETURN UTL_RAW.CAST_TO_NVARCHAR2(DECRYPTED_RAW);
    END DECRYPT_NHANVIEN_LUONG;

```

```

FUNCTION ENCRYPT_NHANVIEN_PHUCAP(
  P_IN IN NVARCHAR2,
  P_KEY IN CHAR
) RETURN RAW DETERMINISTIC IS
  ENCRYPTED_RAW RAW(2000);
BEGIN
  ENCRYPTED_RAW := DBMS_CRYPTO.ENCRYPT( SRC => UTL_RAW.CAST_TO_RAW(P_IN), TYP => ENCRYPTION_TYPE, KEY => UTL_RAW.CAST_TO_RAW(P_KEY) );
  RETURN ENCRYPTED_RAW;
END ENCRYPT_NHANVIEN_PHUCAP;
FUNCTION DECRYPT_NHANVIEN_PHUCAP(
  P_IN IN RAW,
  P_KEY IN CHAR
) RETURN NVARCHAR2 DETERMINISTIC IS
  DECRYPTED_RAW RAW(2000);
BEGIN
  DECRYPTED_RAW := DBMS_CRYPTO.DECRYPT( SRC => P_IN, TYP => ENCRYPTION_TYPE, KEY => UTL_RAW.CAST_TO_RAW(P_KEY) );
  RETURN UTL_RAW.CAST_TO_NVARCHAR2(DECRYPTED_RAW);
END DECRYPT_NHANVIEN_PHUCAP;

```

- Kết quả mã hóa:
- Dữ liệu trước khi mã hóa

| MANV     | TENNV             | PHAI      | NGAYSINH | DIACHI     | SODT | LUONG | PHUCAP          | VAITRO | MANQL | MAPB |
|----------|-------------------|-----------|----------|------------|------|-------|-----------------|--------|-------|------|
| 1 NV131  | Nhân viên 131 Nam | 04-AUG-06 | HCM 131  | 0123456789 | 1014 | 1437  | Nhân viên QL008 | PB004  |       |      |
| 2 NV132  | Nhân viên 132 Nữ  | 25-AUG-83 | HCM 132  | 0123456789 | 1873 | 1989  | Nhân viên QL008 | PB004  |       |      |
| 3 NV133  | Nhân viên 133 Nam | 07-DEC-23 | HCM 133  | 0123456789 | 1578 | 1286  | Nhân viên QL008 | PB004  |       |      |
| 4 NV134  | Nhân viên 134 Nữ  | 17-JUN-87 | HCM 134  | 0123456789 | 1648 | 1487  | Nhân viên QL008 | PB004  |       |      |
| 5 NV135  | Nhân viên 135 Nam | 10-MAY-83 | HCM 135  | 0123456789 | 1863 | 1634  | Nhân viên QL008 | PB004  |       |      |
| 6 NV136  | Nhân viên 136 Nữ  | 09-SEP-23 | HCM 136  | 0123456789 | 1351 | 1623  | Nhân viên QL008 | PB004  |       |      |
| 7 NV137  | Nhân viên 137 Nam | 16-MAY-21 | HCM 137  | 0123456789 | 1256 | 1998  | Nhân viên QL008 | PB004  |       |      |
| 8 NV138  | Nhân viên 138 Nữ  | 29-DEC-94 | HCM 138  | 0123456789 | 1932 | 1086  | Nhân viên QL008 | PB004  |       |      |
| 9 NV139  | Nhân viên 139 Nam | 23-MAR-18 | HCM 139  | 0123456789 | 1943 | 1312  | Nhân viên QL008 | PB004  |       |      |
| 10 NV140 | Nhân viên 140 Nữ  | 12-DEC-05 | HCM 140  | 0123456789 | 1410 | 1158  | Nhân viên QL008 | PB004  |       |      |
| 11 NV141 | Nhân viên 141 Nam | 19-MAR-03 | HCM 141  | 0123456789 | 1606 | 1538  | Nhân viên QL009 | PB005  |       |      |
| 12 NV142 | Nhân viên 142 Nữ  | 06-OCT-14 | HCM 142  | 0123456789 | 1794 | 1080  | Nhân viên QL009 | PB005  |       |      |
| 13 NV143 | Nhân viên 143 Nam | 28-MAR-78 | HCM 143  | 0123456789 | 1741 | 1403  | Nhân viên QL009 | PB005  |       |      |
| 14 NV144 | Nhân viên 144 Nữ  | 18-OCT-84 | HCM 144  | 0123456789 | 1557 | 1369  | Nhân viên QL009 | PB005  |       |      |

- Dữ liệu sau khi mã hóa

| MANV     | TENNV             | PHAI      | NGAYSINH | DIACHI     | SODT                             | LUONG                            | PHUCAP          | VAITRO | MANQL | MAPB |
|----------|-------------------|-----------|----------|------------|----------------------------------|----------------------------------|-----------------|--------|-------|------|
| 1 NV131  | Nhân viên 131 Nam | 04-AUG-06 | HCM 131  | 0123456789 | F494873E51705A969C9589B822D4A9DA | 6BD262AAD8794301376AFB59F8CC8285 | Nhân viên QL008 | PB004  |       |      |
| 2 NV132  | Nhân viên 132 Nữ  | 25-AUG-83 | HCM 132  | 0123456789 | 0D9EE9567A92E5F07EA62CC0AD9BA5F4 | B5EEBEEA59C2E3B80DFADA6513C8021B | Nhân viên QL008 | PB004  |       |      |
| 3 NV133  | Nhân viên 133 Nam | 07-DEC-23 | HCM 133  | 0123456789 | E4115C58FCE1E428ECF7AE99A7E94BB8 | 72EE9C3E71CFFBAE579C7575A39A99C6 | Nhân viên QL008 | PB004  |       |      |
| 4 NV134  | Nhân viên 134 Nữ  | 17-JUN-87 | HCM 134  | 0123456789 | BEC07C06EF36004EDFE173C73D2888A1 | 193DD1795863EA9B8865928FF13FFA67 | Nhân viên QL008 | PB004  |       |      |
| 5 NV135  | Nhân viên 135 Nam | 10-MAY-83 | HCM 135  | 0123456789 | 3A2D3370C6F930DD995F3C71BA3B191  | 68A9B8241E278A211705702777008F85 | Nhân viên QL008 | PB004  |       |      |
| 6 NV136  | Nhân viên 136 Nữ  | 09-SEP-23 | HCM 136  | 0123456789 | B86C7E63453776B81057B1ED60FEE423 | D0012CDE65A94A7173309AB3CED3DA64 | Nhân viên QL008 | PB004  |       |      |
| 7 NV137  | Nhân viên 137 Nam | 16-MAY-21 | HCM 137  | 0123456789 | 78D2413C595BB930503923E91E883BEF | 97F50400317A6440FF2D63DFF1B72090 | Nhân viên QL008 | PB004  |       |      |
| 8 NV138  | Nhân viên 138 Nữ  | 29-DEC-94 | HCM 138  | 0123456789 | 570C9336DC11B9D6228E731982E9A785 | D17D653C2453F7ED9E300DB270BC5C8  | Nhân viên QL008 | PB004  |       |      |
| 9 NV139  | Nhân viên 139 Nam | 23-MAR-18 | HCM 139  | 0123456789 | 5A794731B20B6C85477A2EEA7E1A696C | B204F31B843F3B98817E31336C73D2D5 | Nhân viên QL008 | PB004  |       |      |
| 10 NV140 | Nhân viên 140 Nữ  | 12-DEC-05 | HCM 140  | 0123456789 | 8CC5DBB341C75A4D90FA85E1804261E5 | E7A98CA6BD0053D921273FF49A96017E | Nhân viên QL008 | PB004  |       |      |
| 11 NV141 | Nhân viên 141 Nam | 19-MAR-03 | HCM 141  | 0123456789 | F03B3D9029FA6C399CABE622DC99030D | AB364180FE1D6A5E8671ADB245EE2DE  | Nhân viên QL009 | PB005  |       |      |
| 12 NV142 | Nhân viên 142 Nữ  | 06-OCT-14 | HCM 142  | 0123456789 | DADC6B88F48E91B52F1A46015380013A | 45EBD05ED7C3B28B4A5D21B8FE5741C0 | Nhân viên QL009 | PB005  |       |      |
| 13 NV143 | Nhân viên 143 Nam | 28-MAR-78 | HCM 143  | 0123456789 | 995F2D54C09AE08A4B3BB6C6E7CE19DA | EF5EBDA611FD7FB8475FAD1507245404 | Nhân viên QL009 | PB005  |       |      |
| 14 NV144 | Nhân viên 144 Nữ  | 18-OCT-84 | HCM 144  | 0123456789 | 7AA40AEC2BA5C63F7C22D2326D71BEA  | 4DB9D8F8CBF101FFD10B6A1EAF6A9B79 | Nhân viên QL009 | PB005  |       |      |

### 3. Nhãn an toàn - Oracle Label Security

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

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.

### **Các bước gắn nhãn cho các dòng dữ liệu:**

**Bước 0:** Bởi vì chính sách OLS chỉ được tạo trên PDB nên nếu chưa tạo PDB thì người dùng cần tạo PDB bằng lệnh create pluggable database :

```
ALTER SYSTEM SET db_create_file_dest = 'E:\';
create pluggable database PDB1 admin user QLDA_OLS identified by admin123;
ALTER PLUGGABLE DATABASE PDB1 OPEN;
--ALTER PLUGGABLE DATABASE CLOSE IMMEDIATE;
ALTER SESSION SET CONTAINER = PDB1;
```

**Bước 1:** Kích hoạt OLS bằng 2 lệnh bên dưới vì OLS mặc định không bật khi mới tải về, bắt buộc phải kích hoạt.

```
EXEC LBACSYS.CONFIGURE_OLS;
EXEC LBACSYS.OLS_ENFORCEMENT.ENABLE_OLS;
```

Sau đó tắt đi, khởi động lại SQL Developer, chạy lệnh `select name, status, description from dba_ols_status;` nếu câu truy vấn trả về kết quả này thì kích hoạt OLS thành công

| NAME                   | STATUS | DESCRIPTION                           |
|------------------------|--------|---------------------------------------|
| 1 OLS_CONFIGURE_STATUS | TRUE   | Determines if OLS is configured       |
| 2 OLS_DIRECTORY_STATUS | FALSE  | Determines if OID is enabled with OLS |
| 3 OLS_ENABLE_STATUS    | TRUE   | Determines if OLS is enabled          |

**Bước 2:** Tạo user tạo bảng THONGBAO và chính sách OLS (ở đây là QLDA\_PDB1) và tạo số lượng user đủ dùng.

```

CREATE USER QLDA_PDB1 IDENTIFIED BY admin123;
GRANT CREATE SESSION TO QLDA_PDB1;
GRANT CREATE TABLE, UNLIMITED TABLESPACE TO QLDA_PDB1;
GRANT INHERIT PRIVILEGES ON USER SYS TO QLDA_PDB1;
--GRANT SA_SYSDBA TO QLDA_PDB1;

--Tạo 5 user này trên cả CDB$ROOT và PDB1
CREATE USER TP001 IDENTIFIED BY TP001; --Trưởng phòng phụ trách lĩnh vực sản xuất miền Nam (câu a)
CREATE USER TP002 IDENTIFIED BY TP002; --Trưởng phòng phụ trách tất cả các lĩnh vực không phân biệt chi nhánh (câu b).
CREATE USER TP003 IDENTIFIED BY TP003; --Trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung (câu c).
CREATE USER GD001 IDENTIFIED BY GD001; --Giám đốc có thể xem toàn bộ dữ liệu (câu a)
CREATE USER GD002 IDENTIFIED BY GD002; --Giám đốc phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Bắc (câu a)
CREATE USER NV001 IDENTIFIED BY NV001; --Nhân viên miền trung phụ trách tất cả lĩnh vực (thêm)
--CREATE USER NV002 IDENTIFIED BY NV002; --Nhân viên phụ trách lĩnh vực gia công miền nam (thêm)

```

**Bước 3:** Cấp quyền tạo Session đọc bảng THONGBAO cho các user vừa tạo:

```

GRANT CREATE SESSION TO TP001;
GRANT CREATE SESSION TO TP002;
GRANT CREATE SESSION TO TP003;
GRANT CREATE SESSION TO GD001;
GRANT CREATE SESSION TO GD002;
GRANT CREATE SESSION TO NV001;
GRANT CREATE SESSION TO NV002;

```

**Bước 4:** Tạo chính sách OLS và cấp quyền tương ứng cho user QLDA\_PDB1.

```

EXECUTE SA_SYSDBA.CREATE_POLICY('OLS_QLDA', 'OLS_THONGBAO', 'NO_CONTROL');

GRANT OLS_QLDA_DBA TO QLDA_PDB1;
GRANT EXECUTE ON SA_COMPONENTS TO QLDA_PDB1;
GRANT EXECUTE ON SA_LABEL_ADMIN TO QLDA_PDB1;
GRANT EXECUTE ON SA_POLICY_ADMIN TO QLDA_PDB1;
GRANT EXECUTE ON SA_USER_ADMIN TO QLDA_PDB1;
GRANT EXECUTE ON CHAR_TO_LABEL TO QLDA_PDB1;

```

**Bước 5:** Trên user QLDA\_PDB1, tạo bảng thông báo và thêm các dòng dữ liệu vào bảng.

```

CREATE TABLE THONGBAO (
    MaTB INT,
    NoiDung NVARCHAR2(100),
    ThoiGian TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    DiaDiem NVARCHAR2(50),
    CONSTRAINT PK_THONGBAO PRIMARY KEY (MaTB)
);
--DROP TABLE THONGBAO;
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (1, 'Đây là thông báo cho trưởng phòng phụ trách lĩnh vực sản xuất miền Nam', 'Miền Nam');
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (2, 'Đây là thông báo cho trưởng phòng phụ trách bất kỳ lĩnh vực không phân biệt chi nhánh', NULL);
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (3, 'Đây là thông báo cho trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung', 'Miền Trung');
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (4, 'Đây là thông báo cho giám đốc có thể xem toàn bộ dữ liệu', NULL);
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (5, 'Đây là thông báo cho giám đốc phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Bắc', 'Miền Bắc');
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (6, 'Đây là thông báo cho trưởng phòng phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Nam', 'Miền Nam');
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (7, 'Đây là thông báo cho nhân viên phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Trung', 'Miền Trung');
INSERT INTO THONGBAO (MaTB, NoiDung, DiaDiem) VALUES (8, 'Đây là thông báo cho nhân viên phụ trách lĩnh vực gia công không phân biệt chi nhánh', NULL);

```

**Bước 6:** Cấp quyền đọc bảng THONGBAO cho các user.



```
GRANT SELECT ON QLDA_PDB1.THONGBAO TO TP001;
GRANT SELECT ON QLDA_PDB1.THONGBAO TO TP002;
GRANT SELECT ON QLDA_PDB1.THONGBAO TO TP003;
GRANT SELECT ON QLDA_PDB1.THONGBAO TO GD001;
GRANT SELECT ON QLDA_PDB1.THONGBAO TO GD002;
GRANT SELECT ON QLDA_PDB1.THONGBAO TO NV001;
GRANT SELECT ON QLDA_PDB1.THONGBAO TO NV002;
```

**Bước 7:** Tạo level, compartment và group theo đề bài. Theo đề bài thì chính sách OLS sẽ chia ra 3 level: Giám đốc > Trưởng phòng > Nhân viên; 3 compartment là 3 lĩnh vực: Mua bán, sản xuất, gia công; 3 group là 3 miền: Bắc, Trung, Nam.

```
EXECUTE SA_COMPONENTS.CREATE_LEVEL('OLS_QLDA', 300, 'GD', 'GIAM_DOC');
EXECUTE SA_COMPONENTS.CREATE_LEVEL('OLS_QLDA', 200, 'TP', 'TRUONG_PHONG');
EXECUTE SA_COMPONENTS.CREATE_LEVEL('OLS_QLDA', 100, 'NV', 'NHAN_VIEN');
/*
EXECUTE SA_COMPONENTS.DROP_LEVEL('OLS_QLDA', 300)
EXECUTE SA_COMPONENTS.DROP_LEVEL('OLS_QLDA', 200)
EXECUTE SA_COMPONENTS.DROP_LEVEL('OLS_QLDA', 100)
*/
--Xem owner của nhãn
SELECT * FROM SYS.SA_LABELS;

EXECUTE SA_COMPONENTS.CREATE_COMPARTMENT('OLS_QLDA', 50, 'MB', 'MUA_BAN');
EXECUTE SA_COMPONENTS.CREATE_COMPARTMENT('OLS_QLDA', 40, 'SX', 'SAN_XUAT');
EXECUTE SA_COMPONENTS.CREATE_COMPARTMENT('OLS_QLDA', 30, 'GC', 'GIA_CONG');
/*
EXECUTE SA_COMPONENTS.DROP_COMPARTMENT('OLS_QLDA', 50);
EXECUTE SA_COMPONENTS.DROP_COMPARTMENT('OLS_QLDA', 40);
EXECUTE SA_COMPONENTS.DROP_COMPARTMENT('OLS_QLDA', 30);
*/

EXECUTE SA_COMPONENTS.CREATE_GROUP('OLS_QLDA', 500, 'B', 'MIEN_BAC');
EXECUTE SA_COMPONENTS.CREATE_GROUP('OLS_QLDA', 450, 'T', 'MIEN_TRUNG');
EXECUTE SA_COMPONENTS.CREATE_GROUP('OLS_QLDA', 400, 'N', 'MIEN_NAM');
```

**Bước 8:** Tạo các nhãn cần thiết để gán cho các dòng dữ liệu.

```
--Nhấn giám đốc đọc tất cả dữ liệu
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 1000, 'GD');
--Nhấn giám đốc đọc dữ liệu miền Bắc
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 990, 'GD::B');
--Nhấn giám đốc đọc dữ liệu miền Trung
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 980, 'GD::T');
--Nhấn giám đốc đọc dữ liệu miền Nam
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 970, 'GD::N');
--Nhấn trưởng phòng sản xuất miền Bắc
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 900, 'TP: SX:B');
--Nhấn trưởng phòng sản xuất miền Trung
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 890, 'TP: SX:T');
--Nhấn trưởng phòng sản xuất miền Nam
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 880, 'TP: SX:N');
--Nhấn trưởng phòng
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 870, 'TP');
--Nhấn trưởng phòng tất cả lĩnh vực chi nhánh miền Nam
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 860, 'TP::N');
--Nhấn nhân viên miền trung
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 800, 'NV::T');
--Nhấn nhân viên gia công khu vực miền bắc
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 790, 'NV:GC:B');
--Nhấn nhân viên gia công
EXECUTE SA_LABEL_ADMIN.CREATE_LABEL('OLS_QLDA', 780, 'NV:GC');
```

**Bước 9:** Để gán nhãn cho dữ liệu, ta cần gán chính sách OLS vừa tạo cho bảng THONGBAO nhưng TABLE\_OPTIONS là NULL để chưa áp dụng chính sách OLS.

```
BEGIN
SA_POLICY_ADMIN.APPLY_TABLE_POLICY(
    POLICY_NAME => 'OLS_QLDA',
    SCHEMA_NAME => 'QLDA_PDB1',
    TABLE_NAME => 'THONGBAO',
    TABLE_OPTIONS => NULL
);
END;
```

**Bước 10:** Cập nhật nhãn cho các dòng dữ liệu.

```
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'TP: SX:N') WHERE MaTB=1;
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'TP') WHERE MaTB=2;
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'TP: SX:T') WHERE MaTB=3;
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'GD') WHERE MaTB=4;
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'GD::B') WHERE MaTB=5;
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'TP::N') WHERE MaTB=6;
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'NV::T') WHERE MaTB=7;
UPDATE QLDA_PDB1.THONGBAO SET OLS_THONGBAO=CHAR_TO_LABEL('OLS_QLDA', 'NV:GC') WHERE MaTB=8;
```

**Bước 11:** Xóa chính sách cũ, áp dụng chính sách mới một cách đầy đủ hơn lên bảng THONGBAO.

```

BEGIN
  SA_POLICY_ADMIN.REMOVE_TABLE_POLICY(
    POLICY_NAME => 'OLS_QLDA',
    SCHEMA_NAME => 'QLDA_PDB1',
    TABLE_NAME => 'THONGBAO',
    DROP_COLUMN => FALSE
  );
END;
/

BEGIN
  SA_POLICY_ADMIN.APPLY_TABLE_POLICY(
    POLICY_NAME => 'OLS_QLDA',
    SCHEMA_NAME => 'QLDA_PDB1',
    TABLE_NAME => 'THONGBAO',
    TABLE_OPTIONS => 'READ_CONTROL, WRITE_CONTROL, CHECK_CONTROL'
  );
END;
/

```

**Câu a: Hãy gán nhãn cho 03 người dùng trong hệ thống.**

- 01 giám đốc có thể đọc được toàn bộ dữ liệu: ta gán nhãn MAX\_READ\_LABEL và MAX\_WRITE\_LABEL là GD:MB,SX,GC:B,T,N. Có nghĩa là người dùng này là giám đốc, có compartment chứa cả 3 lĩnh vực và có cả 3 group. Khi đọc dữ liệu bảng thông báo, giám đốc này sẽ đọc được toàn bộ dữ liệu.

```

--GD001: Giám đốc (câu a)
BEGIN
  SA_USER_ADMIN.SET_USER_LABELS(
    POLICY_NAME => 'OLS_QLDA',
    USER_NAME => 'GD001',
    MAX_READ_LABEL => 'GD:MB,SX,GC:B,T,N',
    MAX_WRITE_LABEL => 'GD:MB,SX,GC:B,T,N'
  );
END;

```

- 01 trưởng phòng phụ trách lĩnh vực sản xuất miền Nam: ta gán nhãn MAX\_READ\_LABEL và MAX\_WRITE\_LABEL là TP:SX:N. Có nghĩa là người dùng này là trưởng phòng, có compartment là sản xuất và group là miền Nam. Khi đọc dữ liệu bảng thông báo, trưởng phòng có thể đọc dữ liệu có level là trưởng phòng và nhân viên – có compartment chỉ chứa SX hoặc không có compartment – có group là N hoặc không có level.

```

BEGIN
  SA_USER_ADMIN.SET_USER_LABELS(
    POLICY_NAME => 'OLS_QLDA',
    USER_NAME => 'TP001',
    MAX_READ_LABEL => 'TP:SX:N',
    MAX_WRITE_LABEL => 'TP:SX:N'
  );
END;

```



- 01 giám đốc phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Bắc (có thể đọc được toàn bộ dữ liệu theo đúng cấp bậc và không phân biệt lĩnh vực): ta gán nhãn MAX\_READ\_LABEL và MAX\_WRITE\_LABEL là GD:MB,SX,GC:B. Có nghĩa là người dùng này là giám đốc, có compartment là cả 3 lĩnh vực và group là miền Bắc. Khi đọc dữ liệu bảng thông báo, giám đốc này có thể đọc dữ liệu có level là giám đốc, trưởng phòng và nhân viên – có compartment chứa MB hoặc SX hoặc GC hoặc không có compartment – có group là B hoặc không có level.

```
--GD002: Giám đốc miền bắc (câu a)
BEGIN
  SA_USER_ADMIN.SET_USER_LABELS(
    POLICY_NAME => 'OLS_QLDA',
    USER_NAME  => 'GD002',
    MAX_READ_LABEL  => 'GD:MB, SX, GC: B',
    MAX_WRITE_LABEL => 'GD:MB, SX, GC: B'
  );
END;
```

**Câu b: Hãy cho biết cách thức phát tán dòng thông báo t1 đến tất cả trưởng phòng phụ trách tất cả các lĩnh vực không phân biệt chi nhánh.**

- Phase 1: chính sách OLS sẽ kiểm tra xem level của người dùng có lớn hơn hoặc bằng level của dữ liệu không. Ở đây thì **level dữ liệu <= Trưởng phòng**. Nếu điều kiện này đúng thì tới Phase 2, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 2: kiểm tra group của nhãn người dùng có chứa bất kì group nào trong nhãn dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa bất kì group nào hoặc không có group thì trưởng phòng này sẽ đọc được dữ liệu** vì trưởng phòng này không phân biệt chi nhánh nên nhãn của người dùng này sẽ chứa cả B,T,N trong group. Nếu điều kiện này đúng thì tới Phase 3, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 2: kiểm tra compartment của nhãn người dùng có chứa tất cả các compartment của nhãn dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa bất kì compartment nào hoặc không có compartment thì trưởng phòng này sẽ đọc được dữ liệu** vì trưởng phòng này phụ trách tất cả các lĩnh vực nên nhãn của người dùng này sẽ chứa cả MB, GC, SX trong compartment. Nếu điều kiện này đúng thì người dùng sẽ đọc được dữ liệu, nếu sai thì không được truy cập dòng dữ liệu.
- Đây là ví dụ về một trưởng phòng phụ trách tất cả lĩnh vực không phân biệt chi nhánh:

```
--TP002: trưởng phòng phụ trách tất cả các lĩnh vực không phân biệt chi nhánh (câu b).
BEGIN
  SA_USER_ADMIN.SET_USER_LABELS(
    POLICY_NAME => 'OLS_QLDA',
    USER_NAME   => 'TP002',
    MAX_READ_LABEL => 'TP:MB,SX,GC:B,T,N',
    MAX_WRITE_LABEL => 'TP:MB,SX,GC:B,T,N'
  );
END;
```

- Và đây là kết quả khi SELECT bảng thông báo:

BẢNG "THÔNG BÁO"

|   | MATB | NOIDUNG   | THOIGIAN         | DIADIEM    | OLS_THONGBAO |
|---|------|---|------------------|------------|--------------|
| ▶ | 1    | Đây là thông báo cho trưởng phòng phụ trách lĩnh vực sản xuất miền Nam                | 31/05/2023 02:31 | Miền Nam   | 880          |
|   | 2    | Đây là thông báo cho trưởng phòng phụ trách bất kỳ lĩnh vực không phân biệt chi nhánh | 31/05/2023 02:31 |            | 870          |
|   | 3    | Đây là thông báo cho trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung            | 31/05/2023 02:31 | Miền Trung | 890          |
|   | 6    | Đây là thông báo cho trưởng phòng phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Nam  | 31/05/2023 02:31 | Miền Nam   | 860          |
|   | 7    | Đây là thông báo cho nhân viên phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Trung   | 31/05/2023 02:31 | Miền Trung | 800          |
|   | 8    | Đây là thông báo cho nhân viên phụ trách lĩnh vực gia công không phân biệt chi nhánh  | 31/05/2023 02:31 |            | 780          |
| * |      |   |                  |            |              |

**Câu c: Hãy cho biết cách thức phát tán dòng thông báo t2 đến trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung.**

- Phase 1: chính sách OLS sẽ kiểm tra xem level của người dùng có lớn hơn hoặc bằng level của dữ liệu không. Ở đây thì **level dữ liệu <= Trưởng phòng**. Nếu điều kiện này đúng thì tới Phase 2, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 2: kiểm tra group của nhân người dùng có chứa bất kỳ group nào trong nhãn dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa group T hoặc không có group thì trưởng phòng này sẽ đọc được dữ liệu** vì trưởng phòng này ở miền Trung nên nhãn của người dùng này sẽ chứa T trong group. Nếu điều kiện này đúng thì tới Phase 3, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 3: kiểm tra compartment của nhãn người dùng có chứa tất cả các compartment của nhãn dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa compartment là SX hoặc không có compartment thì trưởng phòng này sẽ đọc được dữ liệu** vì trưởng phòng này phụ trách lĩnh vực sản xuất nên nhãn của người dùng này sẽ chứa SX trong compartment. Nếu điều kiện này đúng thì người dùng sẽ đọc được dữ liệu, nếu sai thì không được truy cập dòng dữ liệu.
- Đây là ví dụ về một trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung:

```
--TP003: trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung (câu c).
BEGIN
  SA_USER_ADMIN.SET_USER_LABELS(
    POLICY_NAME => 'OLS_QLDA',
    USER_NAME   => 'TP003',
    MAX_READ_LABEL => 'TP:MX:T',
    MAX_WRITE_LABEL => 'TP:MX:T'
  );
END;
```

- Và đây là kết quả khi SELECT bảng THONGBAO:

| BẢNG "THÔNG BÁO" |      |   |                  |            |              |
|------------------|------|---|------------------|------------|--------------|
|                  | MATB | NOIDUNG   | THOIGIAN         | DIADIEM    | OLS_THONGBAO |
| ▶                | 2    | Đây là thông báo cho trưởng phòng phụ trách bất kỳ lĩnh vực không phân biệt chi nhánh | 31/05/2023 02:31 |            | 870          |
|                  | 3    | Đây là thông báo cho trưởng phòng phụ trách lĩnh vực sản xuất ở miền Trung            | 31/05/2023 02:31 | Miền Trung | 890          |
|                  | 7    | Đây là thông báo cho nhân viên phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Trung   | 31/05/2023 02:31 | Miền Trung | 800          |
| *                |      |   |                  |            |              |

**Câu d: Em hãy cho thêm một số kịch bản phát tán dữ liệu nữa trên mô hình OLS đã cài đặt.**

Kịch bản 1: Phát tán dòng thông báo t3 tới các nhân viên phụ trách tất cả các lĩnh vực ở chi nhánh miền Trung

- Phase 1: chính sách OLS sẽ kiểm tra xem level của người dùng có lớn hơn hoặc bằng level của dữ liệu không. Ở đây thì **level dữ liệu <= Nhân viên**. Nếu điều kiện này đúng thì tới Phase 2, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 2: kiểm tra group của nhân người dùng có chứa bất kỳ group nào trong nhân dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa group T hoặc không có group thì nhân viên này sẽ đọc được dữ liệu** vì nhân viên này ở miền Trung nên nhân của người dùng này sẽ chứa T trong group. Nếu điều kiện này đúng thì tới Phase 3, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 3: kiểm tra compartment của nhân người dùng có chứa tất cả các compartment của nhân dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa bất kỳ compartment nào hoặc không có compartment thì nhân viên này sẽ đọc được dữ liệu** vì nhân viên này phụ trách tất cả các lĩnh vực nên nhân của người dùng này sẽ chứa cả 3 lĩnh vực trong compartment. Nếu điều kiện này đúng thì người dùng sẽ đọc được dữ liệu, nếu sai thì không được truy cập dòng dữ liệu.
- Đây là ví dụ về nhân viên phụ trách tất cả các lĩnh vực ở miền Trung.

```
--NV001: Nhân viên miền trung phụ trách tất cả lĩnh vực (thêm)
```

```
⊞ BEGIN
```

```
SA_USER_ADMIN.SET_USER_LABELS(
  POLICY_NAME => 'OLS_QLDA',
  USER_NAME   => 'NV001',
  MAX_READ_LABEL => 'NV:MB, SX, GC: T',
  MAX_WRITE_LABEL => 'NV:MB, SX, GC: T'
);
```

```
END;
```

- Và đây là kết quả khi SELECT bảng THONGBAO:

| BẢNG "THÔNG BÁO" |      |  |                  |            |              |
|------------------|------|--|------------------|------------|--------------|
|                  | MATB | NOIDUNG  | THOIGIAN         | DIADIEM    | OLS_THONGBAO |
| ▶                | 7    | Đây là thông báo cho nhân viên phụ trách bất kỳ lĩnh vực nào ở chi nhánh miền Trung  | 31/05/2023 02:31 | Miền Trung | 800          |
|                  | 8    | Đây là thông báo cho nhân viên phụ trách lĩnh vực gia công không phân biệt chi nhánh | 31/05/2023 02:31 |            | 780          |
| *                |      |  |                  |            |              |

Kịch bản 2: Phát tán dòng thông báo t4 tới các nhân viên phụ trách lĩnh vực gia công ở chi nhánh miền Nam

- Phase 1: chính sách OLS sẽ kiểm tra xem level của người dùng có lớn hơn hoặc bằng level của dữ liệu không. Ở đây thì **level dữ liệu <= Nhân viên**. Nếu điều kiện này đúng thì tới Phase 2, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 2: kiểm tra group của nhân người dùng có chứa bất kì group nào trong nhân dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa group N hoặc không có group thì nhân viên này sẽ đọc được dữ liệu** vì nhân viên này ở miền Nam nên nhân của người dùng này sẽ chứa N trong group. Nếu điều kiện này đúng thì tới Phase 3, nếu sai thì không được truy cập dòng dữ liệu.
- Phase 3: kiểm tra compartment của nhân người dùng có chứa tất cả các compartment của nhân dòng dữ liệu hay không. Ở đây thì **nếu dòng dữ liệu chứa compartment GC hoặc không có compartment thì nhân viên này sẽ đọc được dữ liệu** vì nhân viên này phụ trách lĩnh vực gia công nên nhân của người dùng này sẽ chứa GC. Nếu điều kiện này đúng thì người dùng sẽ đọc được dữ liệu, nếu sai thì không được truy cập dòng dữ liệu.
- Đây là ví dụ về nhân viên phụ trách lĩnh vực gia công ở miền Nam:

```
--NV002: Nhân viên phụ trách lĩnh vực gia công miền nam (thêm)
BEGIN
  SA_USER_ADMIN.SET_USER_LABELS(
    POLICY_NAME => 'OLS_QLDA',
    USER_NAME   => 'NV002',
    MAX_READ_LABEL => 'NV:GC:N',
    MAX_WRITE_LABEL => 'NV:GC:N'
  );
END;
```

- Và đây là kết quả khi SELECT bảng THONGBAO:

|   | MATB | NOIDUNG  | THOIGIAN         | DIADIEM | OLS_THONGBAO |
|---|------|--|------------------|---------|--------------|
| ▶ | 8    | Đây là thông báo cho nhân viên phụ trách lĩnh vực gia công không phân biệt chi nhánh | 31/05/2023 02:31 |         | 780          |
| * |      |  |                  |         |              |

#### 4. Ghi vết hệ thống – Audit

Auditing là hoạt động theo dõi và lưu vết lại các hoạt động thao tác của người dùng vào dữ liệu. Trong Oracle, người quản trị có thể cấu hình để thực hiện audit lại các hoạt động trong của cả người dùng trong cơ sở dữ liệu lẫn những người dùng không có trong cơ sở dữ liệu, giới hạn audit với một số lệnh cụ thể hay audit một số role cụ thể trong dữ liệu.

##### Các bước cài đặt Audit:

##### Câu a: Những người đã cập nhật trường THOIGIAN trong quan hệ PHANCONG.

- Ta sử dụng hàm DBMS\_FGA để theo dõi trên quan hệ phân công với chính sách như sau:

```

BEGIN
    DBMS_FGA.ADD_POLICY(
        object_schema => 'QLDA',
        object_name    => 'QLDA_PHANCONG',
        policy_name    => 'THOIGIAN_PHANCONG_AUDIT',
        audit_column    => 'THOIGIAN',
        audit_condition => NULL,
        statement_types => 'UPDATE',
        audit_trail => dbms_fga.db + dbms_fga.extended);
END;

```

Chính sách trên sẽ ghi lại những người đã thực hiện việc UPDATE trong table QLDA\_PHANCONG.

### **Câu b: Những người đã đọc trên trường LUONG và PHUCAP của người khác.**

- Bước 1: Tạo hàm check user audit để kiểm tra xem user dùng view hay dùng table (vì có một số user chỉ được quyền select view). Hàm sẽ trả về 3 nếu đó là QLDA(admin), trả về 1 nếu là người dùng có role 'NV', 'TA', 'GD', 'TC'. Trả về 2 nếu đó là người dùng có role 'QL', 'TP', 'NS'. Và trả về 0 với những người dùng còn lại.

```

CREATE OR REPLACE FUNCTION AUD_F_TABLE_NV(pTxtUser IN VARCHAR2)
RETURN PLS_INTEGER
AS
    USERROLE VARCHAR2(20);
BEGIN
    IF(pTxtUser = 'QLDA') THEN
        RETURN 1;
    END IF;

    SELECT GRANTED_ROLE INTO USERROLE FROM DBA_ROLE_PRIVS WHERE GRANTEE = pTxtUser;

    IF('NV' IN (USERROLE) OR 'TA' IN (USERROLE) OR 'GD' IN (USERROLE) OR 'TC' IN (USERROLE)) THEN
        RETURN 1;
    ELSIF('QL' IN (USERROLE) OR 'TP' IN (USERROLE) OR 'NS' IN (USERROLE)) THEN
        RETURN 2;
    ELSE
        RETURN 0;
    END IF;
END;

```

- Bước 2: thiết lập audit dùng hàm DBMS\_FGA đối với 2 view là V\_QLDA\_NHANVIEN và V\_QLDA\_NHANVIEN\_NS.
  - o Đối với view V\_QLDA\_NHANVIEN: Thiết lập theo dõi và lưu vết đối với những người dùng có role "TA", "NV", "GD", "TC" nhưng đọc LUONG và PHUCAP của người khác. Chính sách như sau:

```

--View V_QLDA_NHANVIEN
begin
    dbms_fga.add_policy(
        object_schema => 'QLDA',
        object_name => 'V_QLDA_NHANVIEN',
        policy_name => 'AUDIT_SELECT_LUONG_PHUCAP',
        audit_column => 'LUONG, PHUCAP',
        audit_condition => 'MANV != USER AND (QLDA.AUD_F_TABLE_NV(USER) = 1 OR QLDA.AUD_F_TABLE_NV(USER) = 0)',
        handler_schema => NULL,
        handler_module => NULL,
        statement_types => 'SELECT',
        --audit_column_opts => dbms_fga.all_columns,
        audit_trail => dbms_fga.db + dbms_fga.extended);
end;

```



Chính sách sẽ ghi viết theo yêu cầu bằng cách thiết lập điều kiện audit\_condition kiểm tra mã nhân viên của trường được SELECT khác với mã nhân viên của người dùng và hàm check user trả về 0 hoặc 1.

- Đối với V\_QLDA\_NHANVIEN\_NS: Thiết lập theo dõi và lưu vết đối với người dùng có role “QL”, “TP”, “NS” nhưng đọc LUONG và PHUCAP của người khác. Chính sách như sau:

```
--Audit view V_QLDA_NHANVIEN_NS
begin
  dbms_fga.add_policy(
    object_schema => 'QLDA',
    object_name => 'V_QLDA_NHANVIEN_NS',
    policy_name => 'AUDIT_SELECT_V_LUONG_PHUCAP',
    audit_column => 'LUONG, PHUCAP',
    audit_condition => 'MANV != USER AND (QLDA.AUD_F_TABLE_NV(USER) = 2 OR QLDA.AUD_F_TABLE_NV(USER) = 0)',
    handler_schema => NULL,
    handler_module => NULL,
    statement_types => 'SELECT',
    --audit_column_opts => dbms_fga.all_columns,
    audit_trail => dbms_fga.db + dbms_fga.extended);
end;
```

Chính sách sẽ ghi vết theo yêu cầu bằng cách thiết lập điều kiện audit\_condition kiểm tra mã nhân viên của trường được SELECT khác với mã nhân viên của người dùng và hàm check user trả về 0 hoặc 2.

### **Câu c: Một người không thuộc vai trò “Tài chính” nhưng đã cập nhật thành công trên trường LUONG và PHUCAP.**

- Bước 1: Kiểm tra xem có phải người dùng thuộc vai trò “Tài chính” không. Ta sẽ tạo hàm để thực hiện việc này. Kết quả trả về của hàm là 1 thì người dùng thuộc vai trò “Tài chính” (không bị audit), trả về 0 thì người dùng không thuộc vai trò “Tài chính” (bị audit).

```
--Hàm kiểm tra role TC, nếu 1 là không bị audit, 0 là bị audit
CREATE OR REPLACE FUNCTION CHECK_TC
RETURN PLS_INTEGER
AS
  USERROLE VARCHAR2(20);
BEGIN
  IF(USER = 'QLDA') THEN
    RETURN 1;
  END IF;

  SELECT GRANTED_ROLE INTO USERROLE FROM DBA_ROLE_PRIVS WHERE GRANTEE = SYS_CONTEXT('userenv', 'SESSION_USER');

  IF('TC' IN (USERROLE)) THEN
    RETURN 1;
  ELSE
    RETURN 0;
  END IF;
END;
```

- Bước 2: Viết audit trên view V\_QLDA\_NHANVIEN với chính sách như sau:

```

begin
    dbms_fga.add_policy(
        object_schema => 'QLDA',
        object_name => 'V_QLDA_NHANVIEN',
        policy_name => 'AUDIT_UPDATE_LUONG_PHUCAP',
        audit_column => 'LUONG, PHUCAP',
        audit_condition => '(QLDA.CHECK_TC) = 0',
        handler_schema => NULL,
        handler_module => NULL,
        statement_types => 'UPDATE',
        --audit_column_opts => dbms_fga.all_columns,
        audit_trail => dbms_fga.db + dbms_fga.extended);
end;
/

```

Chính sách này sẽ ghi vết lại những người dùng thực hiện thao tác UPDATE trên 2 cột LUONG, PHUCAP nhưng không thuộc vai trò “Tài chính” bằng cách thiết lập audit\_condition là hàm kiểm tra trả về 0 và audit\_column là ‘LUONG, PHUCAP’.

- Bước 3: Viết audit trên view V\_QLDA\_NHANVIEN\_NS với chính sách như sau:

```

--View V_QLDA_NHANVIEN_NS (dành cho role QL, TP, NS)
begin
    dbms_fga.add_policy(
        object_schema => 'QLDA',
        object_name => 'V_QLDA_NHANVIEN_NS',
        policy_name => 'AUDIT_V_UPDATE_LUONG_PHUCAP',
        audit_column => 'LUONG, PHUCAP',
        audit_condition => 'QLDA.CHECK_TC = 0',
        handler_schema => NULL,
        handler_module => NULL,
        statement_types => 'UPDATE',
        --audit_column_opts => dbms_fga.all_columns,
        audit_trail => dbms_fga.db + dbms_fga.extended);
end;
/

```

Chính sách này sẽ ghi vết lại những người dùng thực hiện thao tác UPDATE trên 2 cột LUONG, PHUCAP nhưng không thuộc vai trò “Tài chính” bằng cách thiết lập audit\_condition là hàm kiểm tra trả về 0 và audit\_column là ‘LUONG, PHUCAP’.

#### **Câu d: Kiểm tra nhật ký hệ thống.**

- Kiểm tra nhật ký hệ thống với audit của câu a:

```

select AUDIT_TYPE, DBUSERNAME, EVENT_TIMESTAMP, ACTION_NAME, OBJECT_NAME, SQL_TEXT, FGA_POLICY_NAME, OBJECT_TYPE
from unified_audit_trail
where FGA_POLICY_NAME = 'THOIGIAN_PHANCONG_AUDIT' and OBJECT_NAME = 'QLDA_PHANCONG';

```

- Kiểm tra nhật ký hệ thống với audit của câu b:

```

select AUDIT_TYPE, DBUSERNAME, EVENT_TIMESTAMP, ACTION_NAME, OBJECT_NAME, SQL_TEXT, FGA_POLICY_NAME, OBJECT_TYPE
from unified_audit_trail
where FGA_POLICY_NAME = 'AUDIT_SELECT_LUONG_PHUCAP' OR FGA_POLICY_NAME = 'AUDIT_SELECT_V_LUONG_PHUCAP' OR FGA_POLICY_NAME = 'AUDIT_SELECT_LUONG_OTHER';

```

- Kiểm tra nhật ký hệ thống với audit của câu c:

```
select AUDIT_TYPE, DBUSERNAME, EVENT_TIMESTAMP, ACTION_NAME, OBJECT_NAME, SQL_TEXT, FGA_POLICY_NAME, OBJECT_TYPE
from unified_audit_trail
where FGA_POLICY_NAME = 'AUDIT_UPDATE_LUONG_PHUCAP' OR FGA_POLICY_NAME = 'AUDIT_V_UPDATE_LUONG_PHUCAP' OR FGA_POLICY_NAME = 'AUDIT_UPDATE_LUONG_PHUCAP_OTHER';
```

- Kiểm tra tất cả nhật ký hệ thống:

```
select AUDIT_TYPE, DBUSERNAME, EVENT_TIMESTAMP, ACTION_NAME, OBJECT_NAME, SQL_TEXT, FGA_POLICY_NAME, OBJECT_TYPE
from unified_audit_trail;
```

### III. Tài liệu tham khảo

- Các slide lý thuyết + thực hành môn ATBM trong HTTT của trường ĐH KHTN.
- Understanding Oracle Label Security - <https://youtu.be/o4-XpUQWfaM>.
- [DBMS\\_CRYPTO \(oracle.com\)](https://docs.oracle.com/en/database/oracle/oracle-database/21/arpls/DBMS_CRYPT.html#GUID-4B200807-A740-4A2E-8828-AC0CFF6127D5) - [https://docs.oracle.com/en/database/oracle/oracle-database/21/arpls/DBMS\\_CRYPT.html#GUID-4B200807-A740-4A2E-8828-AC0CFF6127D5](https://docs.oracle.com/en/database/oracle/oracle-database/21/arpls/DBMS_CRYPT.html#GUID-4B200807-A740-4A2E-8828-AC0CFF6127D5)
- [TRẦN VĂN BÌNH MASTER: Các câu lệnh hay dùng với Oracle Auditing \(tranvanbinh.vn\)](https://www.tranvanbinh.vn/2022/03/cac-cau-lenh-hay-dung-voi-oracle.html) - <https://www.tranvanbinh.vn/2022/03/cac-cau-lenh-hay-dung-voi-oracle.html>