ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

AN TOÀN VÀ BẢO MẬT TRONG HỆ THỐNG THÔNG TIN

BÁO CÁO ĐỒ ÁN CUỐI KÌ

NHÓM 22

1412101 – Võ Minh Duy

1412689 – Hoàng Thị Bích Vân

MỤC LỤC

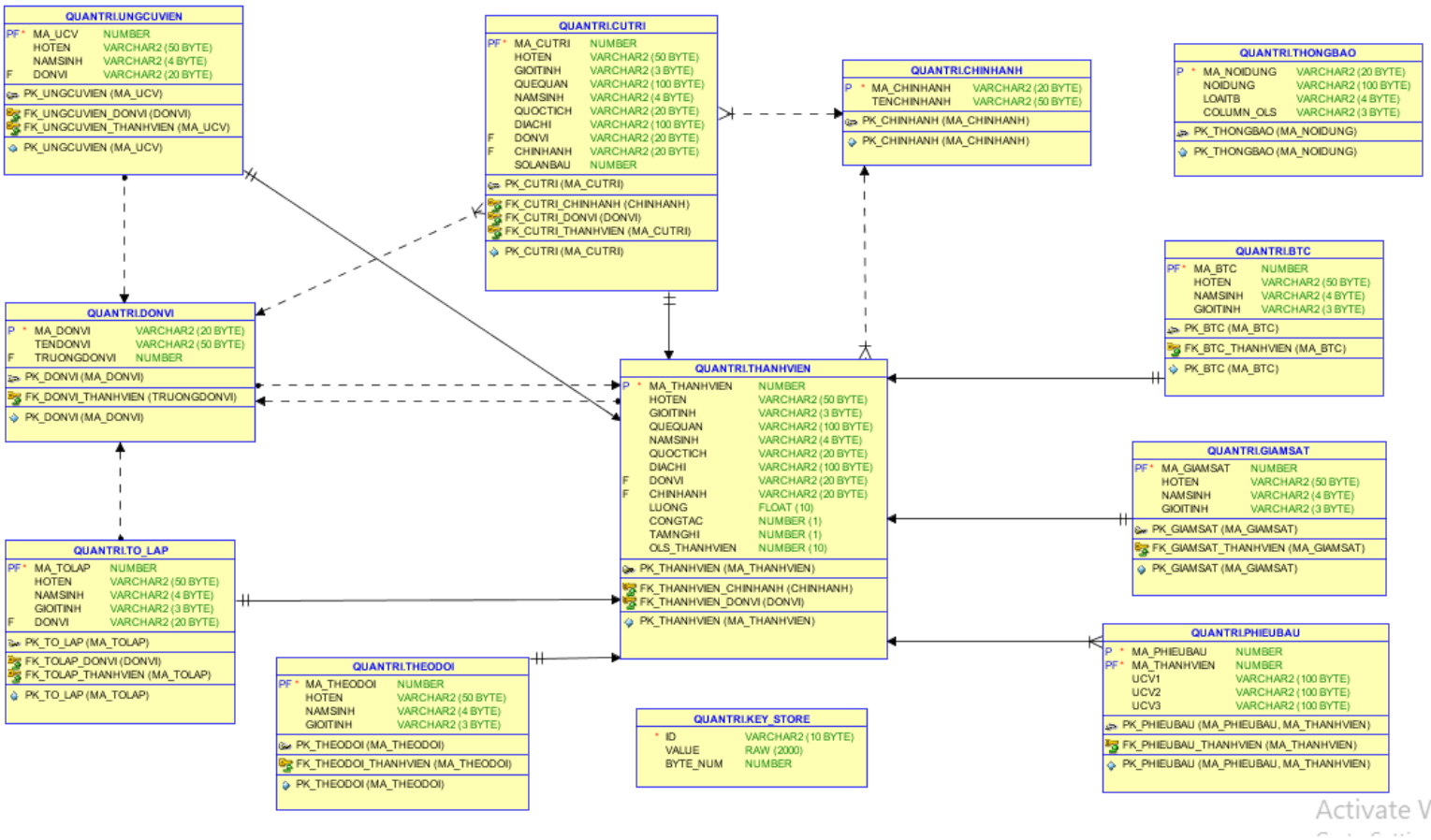
1. Đánh giá đồ án:

STT	Nội dung	Đánh giá
1	**PHẦN 1: HỆ THỐNG DÀNH CHO NGƯỜI QUẢN TRỊ BẢO MẬT** **Yêu cầu: Xây dựng giao diện cho phép người quản trị**	
1.1	Xem danh sách các đối tượng hiện có trên CSDL (user, role, table, view, ….)	
1.2	Thêm mới đối tượng (table, user, role, …)	
1.3	Phân quyền/ lấy lại quyền của một user/ role.	
1.4	Xem quyền của một chủ thể cụ thể.	
2	**PHẦN 2: HIỆN THỰC CÁC CHÍNH SÁCH BẢO MẬT**	
2.1	Báo cáo: Lược đồ CSDL sẽ dùng, liệt kê chính sách bảo mật, phân tích và phân loại. Liệt kê đầy đủ nhưng chỉ cần phân tích và cài đặt **tối thiểu mỗi cơ chế 2 chính sách.**	
2.2	DAC + RBAC	
2.3	VPD	
2.4	MAC (chỉ cần cài đặt 1 chính sách)	
2.5	Mã hóa (chỉ cần cài đặt 1 chính sách)	
2.6	Audit	
2.7	Giao diện cho ứng dụng liên quan các chính sách được cài đặt bởi các cơ chế bảo mật trên.	

2. Thiết kế cơ sở dữ liệu:

2.1. Mô hình quan hệ

2.2. Lược đồ CSDL:



2.3. Access Controls:

2.3.1. DAC:

STT	Role	Quyền	Cài đặt
1	Nhân viên	• Tạo kết nối vào CSDL • Truy vấn bảng THANHVIEN	✔
2	Ban tổ chức	• Truy vấn, thêm, cập nhật bảng UNGCUVIEN,GIAMSAT,TO_LAP,THEDOI	✔
3	Tổ lập	• Truy vấn, thêm, xóa, cập nhật bảng CUTRI	✔

4	Giám sát	• Truy vấn tất cả các bảng trong CSDL	✔
5	Theo dõi kết quả	• Truy vấn bảng PHIEUBAU, CUTRI, UNGVIEN	✔
6	Cử tri	• Thêm, cập nhật bảng PHIEUBAU	✔

- Source-code: /code/oracle_script/access_control.sql

```
/*
    Tạo role trong hệ thống
*/
CREATE ROLE NHANVIEN;
CREATE ROLE BTC;
CREATE ROLE TOLAP;
CREATE ROLE THEODOIKQ;
CREATE ROLE GIAMSAT;
CREATE ROLE CUTRI;

/*
    Cấp quyền cho role NHANVIEN
*/
GRANT CREATE SESSION TO NHANVIEN;
GRANT SELECT ON THANHVIEN TO NHANVIEN;

/*
    Cấp quyền cho role CUTRI
```

```
*/
GRANT INSERT, UPDATE ON PHIEUBAU TO CUTRI;


/*
    Cấp quyền cho role BTC
*/
GRANT SELECT, INSERT, UPDATE ON UNGCUVIEN TO BTC;
GRANT SELECT, INSERT, UPDATE ON GIAMSAT TO BTC;
GRANT SELECT, INSERT, UPDATE ON TO_LAP TO BTC;
GRANT SELECT, INSERT, UPDATE ON THEODOI TO BTC;


/*
    Cấp quyền cho role TOLAP
*/
GRANT SELECT, UPDATE, INSERT, DELETE on CUTRI TO TOLAP


/*
    Cấp quyền cho role THEODOIKQ
*/
GRANT SELECT ON PHIEUBAU TO THEODOIKQ;
GRANT SELECT ON CUTRI TO THEODOIKQ;
GRANT SELECT ON UNGCUVIEN TO THEODOIKQ;
```

```
/*
    Cấp quyền cho role GIAMSAT
*/
GRANT SELECT ON CHINHANH TO GIAMSAT;
GRANT SELECT ON DONVI TO GIAMSAT;
GRANT SELECT ON THANHVIEN TO GIAMSAT;
GRANT SELECT ON BTC TO GIAMSAT;
GRANT SELECT ON TO_LAP TO GIAMSAT;
GRANT SELECT ON GIAMSAT TO GIAMSAT;
GRANT SELECT ON UNGCUVIEN TO GIAMSAT;
GRANT SELECT ON THEODOI TO GIAMSAT;
GRANT SELECT ON THONGBAO TO GIAMSAT;
GRANT SELECT ON PHIEUBAU TO GIAMSAT;
GRANT SELECT ON CUTRI TO GIAMSAT;
```

2.3.2. VPD:

STT	Tên chính sách	Mô tả	Cài đặt
1	Quản lý truy cập thành viên	Thành viên chỉ được truy vấn thông tin của bản thân	✔
2	Quản lý cập nhật phiếu bầu	Cử tri chỉ được phép cập nhật phiếu bầu của bản thân	✔

- Source-code: /code/oracle_script/access_control.sql

```
/*
    Chính sách cho phép :
        NHANVIEN : chỉ xem được thông tin của bản thân
```

```
*/
create or replace FUNCTION f_select_thanhvien (
p_schema VARCHAR2,
p_obj VARCHAR2
)
Return VARCHAR2
AS
    m_user VARCHAR2(128);
    m_donvi VARCHAR2(20);
    user_role VARCHAR2(128);
    stm VARCHAR2(100);
begin
    select user into m_user from dual;
create or replace FUNCTION f_select_thanhvien (
p_schema VARCHAR2,
p_obj VARCHAR2
)
Return VARCHAR2
AS
    m_user VARCHAR2(128);
    m_donvi VARCHAR2(20);
    user_role VARCHAR2(128);
    stm VARCHAR2(100);
```

```
begin
    select user into m_user from dual;


    --Kiểm tra là QUANTRI hoặc người dùng TOLAP
    select granted_role into user_role from
BINHBAU_USER_ROLES where grantee = to_char(m_user) AND
granted_role != 'NHANVIEN';
    if user = 'QUANTRI' or user_role = 'TOLAP' then
        return '';
    end if;
    --Nếu là role khác
    stm := 'MA_THANHVIEN = ' || m_user;
    return stm;
end f_select_thanhvien; end f_select_thanhvien;
BEGIN
    dbms_rls.add_policy (
        object_schema   => 'QUANTRI',
        object_name     => 'THANHVIEN',
        policy_name     => 'Xem_thanhvien_donvi',
        function_schema => 'QUANTRI',
        policy_function => 'f_select_thanhvien_donvi',
        statement_types => 'select'
    );
END;
```

```
/*
    Chính sách cho phép :
        CUTRI : update thông tin phiếu bầu của bản thân
*/
create or replace FUNCTION f_update_cutri_phieubau (
p_schema VARCHAR2,
p_obj VARCHAR2
)
Return VARCHAR2
AS
    m_user VARCHAR2(128);
    stm VARCHAR2(100);
begin
    select user into m_user from dual;
    stm := 'ma_thanhvien = ' || to_char(user);
    return stm;
end f_update_cutri_phieubau;
BEGIN
    dbms_rls.add_policy (
        object_schema   => 'QUANTRI',
        object_name     => 'PHIEUBAU',
        policy_name     => 'Capnhat_phieubau',
        function_schema => 'QUANTRI',
```

```
        policy_function => 'f_update_cutri_phieubau',

        statement_types => 'update'

    );END;
```

2.3.3. OLS:

STT	Tên chính sách	Mô tả	Cài đặt
1	Thanhvien_policy	Gán nhán cho bảng thành viên với các component: • Level: NHANVIEN, TOLAP • Compartment: DONVI(N) • GROUP:CHINHANH(N) Cho phép người dùng TOLAP truy vấn bảng THANHVIEN và lấy những người dùng thuộc cùng đơn vị	✗

- Source-code: /code/oracle_script/access_control.sql

```
execute sa_sysdba.drop_policy('thanhvien_policy');

execute sa_sysdba.create_policy( 'thanhvien_policy',
'ols_thanhvien' );

--Tạo thành phần policy

execute
sa_components.create_level('thanhvien_policy',10,'NV','NHAN
VIEN');

execute
sa_components.create_level('thanhvien_policy',20,'TL','TO
LAP');

execute
sa_components.create_compartment('thanhvien_policy',1,'DV1'
,'DONVI 1');
```

```
execute
sa_components.create_compartment('thanhvien_policy',2,'DV2'
,'DONVI 2');

execute
sa_components.create_compartment('thanhvien_policy',3,'DV3'
,'DONVI 3');

execute
sa_components.create_compartment('thanhvien_policy',4,'DV4'
,'DONVI 4');

execute
sa_components.create_group('thanhvien_policy',100,'CN1','CH
I NHANH 1');

execute
sa_components.create_group('thanhvien_policy',200,'CN2','CH
I NHANH 2');

--Function tự gán nhãn cho dữ liệu

create or replace function tao_nhan_OLS_thanhvien (

    p_mtv in number,

    p_donvi in varchar2,

    p_chinhanh in varchar2

)

return lbacsys.lbac_label as

    v_label varchar2(100);

    m_donvi varchar(20);

    m_chinhanh varchar(20);

begin

    FOR u IN (select granted_role from BINHBAU_USER_ROLES
where grantee = TO_CHAR(p_mtv))
```

```
    LOOP
        if u.granted_role = 'TOLAP' then
            v_label := v_label || 'TL:';
        else
            v_label := 'NV:';
        end if;
    END LOOP;
    if p_donvi = '1' then
        v_label := 'DV1:';
    elsif p_donvi = '2' then
        v_label := 'DV2:';
    elsif p_donvi = '3' then
        v_label := 'DV3:';
    else
        v_label := 'DV4:';
    end if;
    if p_chinhanh = '1' then
        v_label := v_label || 'CN1';
    else
        v_label := v_label || 'CN2';
    end if;

    return TO_LBAC_DATA_LABEL('thanhvien_policy',v_label);
```

```
end tao_nhan_OLS_thanhvien;

BEGIN

    sa_policy_admin.apply_table_policy (

        policy_name     => 'thanhvien_policy',

        schema_name     => 'QUANTRI',

        table_name      => 'THANHVIEN',

        table_options   => '
READ_CONTROL,WRITE_CONTROL,CHECK_CONTROL ',

        label_function  =>
'QUANTRI.tao_nhan_OLS_thanhvien(:new.ma_thanhvien,:new.donv
i,:new_chinhanh)',

        predicate       => NULL

        );

END;
```

2.3.4. Mã hóa:

STT	Mô tả	Hoạt động	Cài đặt
1	Mã hóa thông tin ứng cử viên trên phiếu bầu. Tổ theo dõi chỉ có thể xem thông tin ứng cử viên trên phiếu bầu nhưng không xem được ai bỏ phiếu đó.	1. Trigger ALTER_PHIEUBAU_TRIGGER khi thêm 1 dòng vào bảng PHIEUBAU 2. Tổ theo dõi dùng lệnh: select * from table(query_ungvien()); để xem giải mã.	✔

2	Mã hóa thông tin lương của thành viên.		✕

- Source-code: /code/oracle_script/mahoa.sql

```
--Trigger mã hóa thông tin UCV được thêm vào phiếu bầu

create or replace TRIGGER alter_phieubau_trigger

    BEFORE UPDATE OR INSERT ON PHIEUBAU

    FOR EACH ROW

declare

    ciphertext RAW(2000);

    current_user VARCHAR(50);

    m_key RAW(32);

BEGIN

    select KEY_STORE.value

        into m_key

        from KEY_STORE

        where ID = '1';


    ciphertext := encrypt_data_AES128(  in_data =>
UTL_I18N.STRING_TO_RAW(:new.UCV1,'AL32UTF8'),

                                        in_key  => m_key
```

```
                                        );

    :new.UCV1 := UTL_RAW.CAST_TO_VARCHAR2(ciphertext);


    ciphertext := encrypt_data_AES128(  in_data =>
UTL_I18N.STRING_TO_RAW(:new.UCV2,'AL32UTF8'),

                                        in_key  => m_key

                                        );

    :new.UCV2 := UTL_RAW.CAST_TO_VARCHAR2(ciphertext);


    ciphertext := encrypt_data_AES128(  in_data =>
UTL_I18N.STRING_TO_RAW(:new.UCV3,'AL32UTF8'),

                                        in_key  => m_key

                                        );

    :new.UCV3 := UTL_RAW.CAST_TO_VARCHAR2(ciphertext);

END;


INSERT INTO PHIEUBAU

    VALUES ('1','TDA01','Jacquenetta Jenoure','Annabel
Dunlop','Gisele Grice');


SELECT * FROM PHIEUBAU;


--Tạo function để lấy thông tin phiếu bầu

create or replace TYPE PHIEUBAU_TYPE
```

```
AS OBJECT (
    UCV1 VARCHAR2(100),
    UCV2 VARCHAR2(100),
    UCV3 VARCHAR2(100),
);


create or replace TYPE PHIEUBAU_TABLE
AS TABLE OF PHIEUBAU_TYPE;


CREATE OR REPLACE FUNCTION QUERY_UNGVIEN
RETURN PHIEUBAU_TABLE
PIPELINED
AS
    plaintext VARCHAR2(100);
    ciphertext RAW(2000);
    m_key RAW(32);
    curr_user VARCHAR2(128);
    user_role VARCHAR2(128);
BEGIN
    curr_user := SYS_CONTEXT('userenv','SESSION_USER');
    select granted_role into user_role
        from BINHBAU_USER_ROLES
```

```
        where grantee = curr_user AND granted_role !=
'NHANVIEN';

        exception

        when NO_DATA_FOUND then

            user_role := '';

    IF user_role = 'THEODOIKQ' THEN

        --select key

        select KEY_STORE.value

            into m_key

            from KEY_STORE

            where ID = '1';

        FOR pb IN (select UCV1, UCV2, UCV3 from PHIEUBAU)

        LOOP

            --decrypt UCV1

            pb.UCV1 :=
UTL_RAW.CAST_TO_NUMBER(decrypt_data_AES128(UTL_I18N.STRING_
TO_RAW(pb.UCV1),m_key));

            --decrypt UCV2

            pb.UCV2 :=
UTL_RAW.CAST_TO_NUMBER(decrypt_data_AES128(UTL_I18N.STRING_
TO_RAW(pb.UCV2),m_key));

            --decrypt UCV3

            pb.UCV3 :=
UTL_RAW.CAST_TO_NUMBER(decrypt_data_AES128(UTL_I18N.STRING_
TO_RAW(pb.UCV3),m_key));

        PIPE ROW (PHIEUBAU_TYPE(pb.UCV1,pb.UCV2,pb.UCV3));
```

```
        END LOOP;
    ELSE
        raise_application_error(-20000,'User authority
level is not sufficient');
    END IF;
END;


--Cho phép người dùng THEODOIKQ sử dụng function decrypt
create or replace trigger capquyen_mahoa_phieubau
    AFTER INSERT ON THEODOI
    FOR EACH ROW
DECLARE
    PRAGMA AUTONOMOUS_TRANSACTION;
    stm varchar2(100);
BEGIN
    stm := 'GRANT EXECUTE ON QUANTRI.QUERY_UNGVIEN TO "'
|| :new.ma_theodoi || '"';
    execute immediate stm;
    commit;
END capquyen_mahoa_phieubau;
select * from table(query_ungvien());
```

2.4. Chức năng:

- Procedure : P
- Function: F
- Trigger: T

STT	Loại	Tên	Hoạt động
1	P	CAPNHAT_ROLE	Tự động cập nhật lại Role của người dùng tương ứng với bảng
2	P	TAO_USER	Tạo user trong CSDL Gán quyền NHANVIEN khi thêm 1 dòng dữ liệu vào bảng THANHVIEN Mật khẩu mặc định: 123456
3	F	ENCRYPT_DATA_AES128	Mã hóa thông tin sử dụng thuật toán AES128
4	F	DECRYPT_DATA_AES128	Giải mã thông tin sử dụng thuật toán AES128
5	F	F_SELECT_THANHVIEN	Sử dụng trong VPD 1 Trả về rỗng nếu người dùng là QUANTRI hoặc TOLAP
6	F	F_UPDATE_CUTRI_PHIEUBAU	Sử dụng trong VPD 2
7	F	HASH_DATA	
8	F	QUERY_UNGVIEN	Trả về giải mã thông tin ứng cử viên của bảng PHIEUBAU
9	T	ALTER_PHIEUBAU_TRIGGER	Điều kiện: SELECT | UPDATE bảng PHIEUBAU Mã hóa thông tin ứng cử viên trước khi thêm dữ liệu vào bảng

10	T	CAPQUYEN_MAHOA_PHIEUBAU	Điều kiện: INSERT bảng THEODOI Cấp quyền thực thi chức năng QUERY_PHIEUBAU
12	T	TAO_CONNECTION	Điều kiện: INSERT bảng THANHVIEN Lấy năm hệ thống và thêm vào trước MATHANHVIEN sao đó gọi TAO_USER
13	T	THEM_(BTC|CUTRI|GIAMSAT|THEODOI|TOLAP)	Điều kiện: INSERT bảng (BTC|CUTRI|GIAMSAT|THEODOI|TOLAP) Gọi procedure CAPNHAT_ROLE

3. Giao diện: