ĐẠI HỌC KHOA HỌC TỰ NHIÊN

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

ĐỒ ÁN MÔN HỌC

PHÂN HỆ 1 & 2

MÔN HỌC: AN TOÀN VÀ BẢO MẬT DỮ LIỆU TRONG HTTT

Giảng viên hướng dẫn:

TS. Phạm Thị Bạch Huệ

ThS. Lương Vĩ Minh

ThS. Tiết Gia Hồng

Nhóm thực hiện: 20H3T1 - 08

Lớp: 20HTTT1

Học kỳ II – Năm học: 2022 – 2023

MỤC LỤC

I. THÔNG TIN NHÓM

Mã nhóm: 20H3T1 – 08

Thành viên:

STT	MSSV	Họ và tên
1	20127063	Phan Minh Phúc
2	20127548	Đổng Mỹ Linh
3	20127570	Võ Thị Kim Ngân
4	20127671	Phạm Quốc Vương

Phân công công việc

Phân hệ	MSSV	Công việc
1	20127063	Cài đặt chức năng Đăng nhập
	20127063	Cài đặt chức năng Xem danh sách người dùng trong hệ thống
	20127063	Cài đặt chức năng Thông tin về quyền (privileges) của mỗi user/ role trên các đối tượng dữ liệu
	20127570	Cài đặt chức năng Cho phép tạo mới, xóa, sửa (hiệu chỉnh) user hoặc role
	20127570	Viết thủ tục (storedprocedure) cho việc tạo mới, xóa, sửa user/ role.
	20127671	Cài đặt chức năng Cho phép thực hiện việc cấp quyền cho user, cấp quyền cho role, cấp role cho user. (Cấp quyền thao tác trên đối tượng)
	20127570	Cài đặt chức năng Cho phép thu hồi quyền từ người dùng/ role
	20127671	Cài đặt chức năng Cho phép kiểm tra quyền của các chủ thể vừa được cấp quyền
	20127548	Cài đặt chức năng Cho phép chỉnh sửa quyền của user/ role (Quyền hệ thống)
	20127548	Viết thủ tục (storedprocedure) cho việc chỉnh sửa quyền (Quyền hệ thống)
2	20127671	Xây dựng ứng dụng
		Cài đặt cơ chế bảo mật cho chính sách 1,2,4
	20127548	Cài đặt cơ chế bảo mật cho chính sách 3,5,6
	20127570	Cài đặt mã hóa
	20127063	Cài đặt chính sách Audit

II. BÁO CÁO ĐỒ ÁN

1. PHÂN HỆ 1

- **Các chức năng và tình trạng hoàn thành**

STT	Tên chức năng	Tình trạng
1.	Đăng nhập	Đã hoàn tất
2.	Xem danh sách người dùng trong hệ thống	Đã hoàn tất
3.	Thông tin về quyền (privileges) của mỗi user/ role trên các đối tượng dữ liệu	Đã hoàn tất
4.	Cho phép tạo mới, xóa, sửa (hiệu chỉnh) user hoặc role	Đã hoàn tất
5.	Cho phép thực hiện việc 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. Quyền select, update thì cho phép phân quyền tính đến mức cột; quyền insert, delete thì không	Đã hoàn tất
6.	Cho phép thu hồi quyền từ người dùng/ role	Đã hoàn tất
7.	Cho phép kiểm tra quyền của các chủ thể vừa được cấp quyền	Đã hoàn tất
8.	Cho phép chỉnh sửa quyền của user/ role	Đã hoàn tất

2. PHÂN HỆ 2

2.1 CÁC CHÍNH SÁCH ĐÃ CÀI THEO TỪNG CHÍNH SÁCH BẢO MẬT

- Nhóm đã tạo ra 7 ROLE tương ứng với 7 vai trò trong đề, lần lượt là (tên vai trò- tên role)
    - Nhân viên – NHANVIEN
    - QL trực tiếp – QLNV
    - Trưởng phòng – TRUONGPHONG
    - Tài chính – TAICHINH
    - Nhân sự - NHANSU
    - Trưởng đề án – TRUONGDEAN
    - Ban giám đốc – GIAMDOC
- Sau đó, nhóm thực hiện tạo tài khoản cho mỗi người dùng trong hệ thống và gán các người dùng có vai trò tương ứng vào ROLE đã được tạo.
- Đối với việc gán ROLE, nhóm thực hiện theo cơ chế bảo mật DAC, để gán ROLE cho từng người dùng.
- Đối với các quyền tương ứng với các chính sách bảo mật, nhóm thực hiện theo cơ chế bảo mật RBAC để có thể gán quyền cho từng vai trò tương ứng.

2.1.1 Chính sách 1

- Nội dung: Những người dùng có vai trò là "**Nhân viên**" là một nhân viên thông thường, không kiêm nhiệm công việc nào khác và có các quyền như trong bảng và tương ứng các giải pháp của nhóm

Quyền	Giải pháp
Xem tất cả thuộc tính trên quan hệ NHANVIEN liên quan đến chính nhân viên.	Tạo view **xemthongtincanhan** *CREATE OR REPLACE VIEW XemThongTinCaNhan* *AS* *SELECT * FROM GROUP08.qlnv_nhanvien* *WHERE* *MANV=SYS_CONTEXT('USERENV','SESSION_USER');* - Với điều kiện MANV bằng với USERNAME mà người dùng đăng nhập (sử dụng Hàm SYS_CONTEXT để lấy ra USERNAME của người dùng đang đăng nhập). - Sau đó cấp quyền xem view này cho vai trò Nhân viên (tương ứng là ROLE: NHANVIEN) theo lệnh: GRANT SELECT ON **xemthongtincanhan TO NHANVIEN**
Xem tất cả thuộc tính trên quan hệ PHANCONG liên quan đến chính nhân viên.	Tạo view **XemThongTinPhanCong** *CREATE VIEW XemThongTinPhanCong* *AS* *SELECT * FROM GROUP08.qlnv_phancong* *WHERE* *MANV=SYS_CONTEXT('USERENV','SESSION_USER');* - Với điều kiện MANV bằng với USERNAME mà người dùng đăng nhập (sử dụng Hàm SYS_CONTEXT để lấy ra USERNAME của người dùng đang đăng nhập). - Sau đó, cấp quyền xem view này cho vai trò Nhân viên (tương ứng là ROLE: NHANVIEN) theo lệnh: GRANT SELECT ON **XemThongTinPhanCong TO NHANVIEN**
Có thể sửa trên các thuộc tính NGAYSINH, DIACHI, SODT liên quan đến chính nhân viên đó	- Tạo trigger để người dùng có thể chỉnh sửa quan hệ NHANVIEN thông qua View đã được cấp. *CREATE OR REPLACE TRIGGER* ***CapNhat_TTCaNhan*** *INSTEAD OF* ***UPDATE*** *ON* ***XemThongTinCaNhan*** *FOR EACH ROW* *DECLARE* *BEGIN* *UPDATE GROUP08.qlnv_nhanvien*

	SET NGAYSINH = :NEW.NGAYSINH, DIACHI = :NEW.DIACHI, SODT = :NEW.SODT *WHERE username = :NEW.username;* *END;* - Thực hiện cấp quyền Sửa trên view đã cấp cho vai trò NHANVIEN theo lệnh: GRANT UPDATE(NGAYSINH,DIACHI,SODT) ON **xemthongtincanhan TO NHANVIEN**
Có thể xem dữ liệu của toàn bộ quan hệ PHONGBAN và DEAN	- Thực hiện cấp quyền xem trên quan hệ PHONGBAN và DEAN như sau: o GRANT SELECT ON GROUP08.QLNV_PHONGBAN TO NHANVIEN (Quan hệ PHONGBAN, nhóm đặt lưu trữ tên bảng là *QLNV_PHONGBAN*) o GRANT SELECT ON GROUP08.QLNV_DEAN TO NHANVIEN (Quan hệ PHONGBAN, nhóm đặt lưu trữ tên bảng là *QLNV_PHONGBAN*)

2.1.2 Chính sách 2

- Nội dung: Những người dùng có vai trò là "**QL trực tiếp**" nếu họ có phụ trách quản lý trực tiếp nhân viên khác và có các quyền như trong bảng và tương ứng các giải pháp của nhóm

Quyền	Giải pháp
Có quyền như một nhân viên thông thường	- Thực hiện cấp các quyền tương ứng như đã cấp cho ROLE NVQL, như sau: o GRANT SELECT ON **xemthongtincanhan** to NVQL; o GRANT UPDATE (NGAYSINH,DIACHI,SODT) ON xemthongtincanhan to NVQL; o GRANT SELECT ON GROUP08.XemThongTinPhanCong TO NVQL; o GRANT SELECT ON GROUP08.QLNV_PHONGBAN TO NVQL; o GRANT SELECT ON GROUP08.QLNV_DEAN TO NVQL;

Có quyền xem các dòng trên quan hệ NHANVIEN mà liên quan đến các nhân viên mà mình trực tiếp quản lý (trừ thuộc tính LUONG, PHUCAP)	- Tạo view **NVQL_XemThongTin_NV** *CREATE OR REPLACE VIEW* ***NVQL_XemThongTin_NV*** *AS SELECT MANV,TENNV,PHAI,NGAYSINH,DIACHI,SODT, VAITRO,MANQL,PHG FROM GROUP08.qlnv_nhanvien WHERE MANQL=SYS_CONTEXT('USERENV','SESSION_ USER');* - Với điều kiện MANQL bằng với USERNAME mà người dùng đăng nhập (sử dụng Hàm SYS_CONTEXT để lấy ra USERNAME của người dùng đang đăng nhập). Từ đó, có thể lấy ra những nhân viên do người dùng đang đăng nhập quản lý. - Thực hiện cấp quyền xem trên view này cho ROLE NVQL (tương ứng với người dùng có vai trò QL Trực tiếp) với lệnh: GRANT SELECT ON **NVQL_XemThongTin_NV TO NVQL**
Xem các dòng trong quan hệ PHANCONG liên quan đến chính mình và các nhân viên mà được mình quản lý trực tiếp	- Tạo view **NVQL_Xem_PC** *CREATE OR REPLACE VIEW* ***NVQL_Xem_PC*** *AS SELECT PC.* FROM GROUP08.qlnv_phancong PC JOIN GROUP08.qlnv_NHANVIEN NV ON PC.MANV=NV.MANV WHERE SYS_CONTEXT('USERENV','SESSION_USER')= NV.MANQL OR PC.MANV=SYS_CONTEXT('USERENV','SESSIO N_USER');* - Thực hiện kết 2 bảng NHANVIEN và PHANCONG với điều kiện kết là MANV để lấy ra phân công của mỗi nhân viên. Với điều kiện ở mệnh đề WHERE là chọn ra các dòng mà Nhân viên có MANQL bằng với USERNAME mà người dùng đăng nhập (sử dụng Hàm SYS_CONTEXT để lấy ra USERNAME của người dùng đang đăng

	nhập). Từ đó, có thể lấy ra những nhân viên do người dùng đang đăng nhập quản lý. Sau đó thực hiện cấp quyền xem view này cho ROLE NVQL, như sau: GRANT SELECT ON **NVQL_Xem_PC** TO NVQL

2.1.3 Chính sách 3

- Nội dung: Những người dùng có vai trò là "**Trưởng phòng**" cho biết đó là một nhân viên kiêm nhiệm thêm vai trò trưởng phòng và có các quyền như trong bảng và tương ứng các giải pháp của nhóm

Quyền	Giải pháp
Có quyền như một nhân viên thông thường	- Thực hiện cấp các quyền tương ứng như đã cấp cho ROLE NVQL, như sau: o GRANT SELECT ON **xemthongtincanhan** to TRUONGPHONG; o GRANT UPDATE (NGAYSINH,DIACHI,SODT) ON xemthongtincanhan to TRUONGPHONG; o GRANT SELECT ON GROUP08.XemThongTinPhanCong TO TRUONGPHONG; o GRANT SELECT ON GROUP08.QLNV_PHONGBAN TO TRUONGPHONG; o GRANT SELECT ON GROUP08.QLNV_DEAN TO TRUONGPHONG;
Có thể xem các dòng trong quan hệ NHANVIEN liên quan đến các nhân viên thuộc phòng ban mình làm trưởng phòng (trừ thuộc tính LUONG, PHUCAP)	- Tạo view **TP_XemThongTin_NV** *CREATE OR REPLACE VIEW* ***TP_XemThongTin_NV*** *AS SELECT MANV,TENNV,PHAI,NGAYSINH,DIACHI,SODT, VAITRO,MANQL,PHG FROM GROUP08.qlnv_nhanvien WHERE PHG=(SELECT MAPB FROM GROUP08.qlnv_phongban WHERE TRPHG=SYS_CONTEXT('USERENV','SESSION_ USER'));* Tạo view để người dùng xem các thông tin cần thiết và loại đi hai cột LUONG, PHUCAP. Và với điều kiện ở mệnh đề WHERE là PHG của nhân viên bằng với MAPB (Mã phòng ban – thuộc quan

	hệ PHONGBAN) của người dùng đang đăng nhập và với điều kiện lấy ra MAPB là TRPHG (Mã trưởng phòng) bằng với USERNAME mà người dùng đăng nhập (sử dụng Hàm SYS_CONTEXT để lấy ra USERNAME của người dùng đang đăng nhập). Từ đó, có thể lấy ra những nhân viên thuộc phòng mà người dùng đang đăng nhập làm trưởng phòng. Sau đó, thực hiện cấp quyền xem trên view này: GRANT SELECT ON **TP_XemThongTin_NV** TO TRUONGPHONG
Có thể thêm, xóa, cập nhật trên quan hệ PHANCONG liên quan đến các nhân viên thuộc phòng ban minh làm trưởng phòng	- Tạo view để người dùng có vai trò TRUONGPHONG có thể xem và từ đó chỉnh sửa thông qua view đã tạo, cụ thể như sau: - Tạo view để người dùng xem: **TP_XemPC_NV** *CREATE OR REPLACE VIEW **TP_XemPC_NV*** *AS* *SELECT PC.MANV, PC.MADA, PC.THOIGIAN* *FROM GROUP08.qlnv_phancong PC* *JOIN GROUP08.qlnv_NHANVIEN NV ON PC.MANV=NV.MANV* *WHERE PC.MANV IN (SELECT MANV FROM GROUP08.QLNV_NHANVIEN WHERE PHG=(SELECT MAPB FROM GROUP08.QLNV_PHONGBAN WHERE TRPHG=SYS_CONTEXT('USERENV','SESSION_USER')))* *--WITH CHECK OPTION;* - Tương tự như trên, điều kiện cũng là lọc ra các dòng phân công thuộc phòng ban mình làm trưởng phòng. Tiếp theo, thực hiện tạo trigger để người dùng có thể chỉnh sửa thông qua view này. - Tạo trigger để cho việc thêm trên view: *CREATE OR REPLACE TRIGGER **TP_ThemPC_NV*** *INSTEAD OF **INSERT** ON **TP_XemPC_NV*** *FOR EACH ROW* *DECLARE PHONGNV VARCHAR2(20);* *PHONGTRPHG VARCHAR2(20);* *BEGIN* *SELECT PHG INTO PHONGNV FROM GROUP08.QLNV_NHANVIEN WHERE MANV=:NEW.MANV;*

	SELECT MAPB INTO PHONGTRPHG FROM GROUP08.QLNV_PHONGBAN WHERE TRPHG=SYS_CONTEXT('USERENV','SESSION_USER'); *IF PHONGNV=PHONGTRPHG THEN* *INSERT INTO GROUP08.qlnv_phancong(MANV,MADA,THOIGIAN)* *VALUES (:new.MANV,:new.MADA,:new.THOIGIAN);* *END IF;* *END;* - Thực hiện khai báo 2 biến là ***PHONGNV*** (Mã phòng của nhân viên được thêm phân công) và ***PHONGTRPHG*** (Mã phòng của người dùng có vai trò Trưởng phòng đang đăng nhập và thực hiện thêm phân công). Sau đó tiến hành kiểm tra nếu 2 mã phòng này khớp với nhau thì mới tiến hành việc thêm phân công vào bảng. - Tiếp theo là tạo trigger cho việc cập nhật phân công *CREATE OR REPLACE TRIGGER* ***TP_CapNhat_NV*** *INSTEAD OF* ***UPDATE*** *ON* ***TP_XemPC_NV*** *FOR EACH ROW* *BEGIN* *UPDATE GROUP08.qlnv_phancong* *SET MANV=:NEW.MANV, MADA=:NEW.MADA, THOIGIAN=:NEW.THOIGIAN* *where manv=:new.manv;* *END;* - Và cuối cùng là tạo trigger cho việc xóa phân công *CREATE OR REPLACE TRIGGER* ***TP_Xoa_NV*** *INSTEAD OF* ***DELETE*** *ON* ***TP_XemPC_NV*** *FOR EACH ROW* *BEGIN*

	DELETE FROM GROUP08.qlnv_phancong where manv=:old.manv; *END;* - Sau đó, cấp thêm quyền Thêm, xóa, cập nhật view này cho người dùng như sau: GRANT INSERT, DELETE, UPDATE ON **TP_XemPC_NV** TO TRUONGPHONG

2.1.4 Chính sách 4

- Nội dung: Những người dùng có vai trò là "**Tài chính**" cho biết đó là một nhân viên phụ trách công tác tài chính tiền lương của công ty và có các quyền như trong bảng và tương ứng các giải pháp của nhóm

Quyền	Giải pháp
Có quyền như một nhân viên thông thường	- Thực hiện cấp các quyền tương ứng như đã cấp cho ROLE NVQL, như sau: o GRANT SELECT ON **xemthongtincanhan** to TAICHINH; o GRANT UPDATE (NGAYSINH,DIACHI,SODT) ON xemthongtincanhan to TAICHINH; o GRANT SELECT ON GROUP08.XemThongTinPhanCong TO TAICHINH; o GRANT SELECT ON GROUP08.QLNV_PHONGBAN TO TAICHINH; o GRANT SELECT ON GROUP08.QLNV_DEAN TO TAICHINH;
Xem trên toàn bộ quan hệ NHANVIEN, PHANCONG,	- Thực hiện cấp quyền đọc trên quan hệ NHANVIEN và PHANCONG theo lệnh : o GRANT SELECT ON GROUP08.QLNV_NHANVIEN TO TAICHINH o GRANT SELECT ON GROUP08.QLNV_PHANCONG TO TAICHINH
Có thể chỉnh sửa trên thuộc tính LUONG và PHUCAP	- Thực hiện cấp quyền chính sửa trên 2 thuộc tính LUONG, PHUCAP trên quan hệ NHANVIEN theo lệnh: GRANT UPDATE (LUONG,PHUCAP) ON GROUP08.QLNV_NHANVIEN TO TAICHINH

2.1.5 Chính sách 5

- Nội dung: Những người dùng có vai trò là "**Nhân sự**" cho biết đó là nhân viên phụ trách công tác nhân sự trong công ty và có các quyền như trong bảng và tương ứng các giải pháp của nhóm

Quyền	Giải pháp
Có quyền như một nhân viên thông thường	- Thực hiện cấp các quyền tương ứng như đã cấp cho ROLE NVQL, như sau: o GRANT SELECT ON **xemthongtincanhan** to NHANSU; o GRANT UPDATE (NGAYSINH,DIACHI,SODT) ON xemthongtincanhan to NHANSU; o GRANT SELECT ON GROUP08.XemThongTinPhanCong TO NHANSU; o GRANT SELECT ON GROUP08.QLNV_PHONGBAN TO TAICHINH; o GRANT SELECT ON GROUP08.QLNV_DEAN TO NHANSU;
Được quyền thêm, cập nhật trên quan hệ PHONGBAN	- Thực hiện cấp quyền thêm, cập nhật trên quan hệ PHONGBAN theo lệnh: GRANT INSERT, UPDATE ON GROUP08.QLNV_PHONGBAN TO NHANSU
Thêm, cập nhật dữ liệu trong quan hệ NHANVIEN với các giá trị các trường LUONG,PHUCAP là mang giá trị mặc định NULL, không được xem LUONG,PHUCAP của người khác và không được cập nhật trên các trường LUONG,PHUCAP	- Thực hiện tạo view để người dùng có thể xem dữ liệu trên quan hệ NHANVIEN và thực hiện thêm, cập nhật thông qua view này. - Đầu tiên, tạo view để người dùng có thể xem trên quan hệ NHANVIEN *CREATE OR REPLACE VIEW* ***NS_XemThongTin_NV*** *AS* *SELECT* *MANV,TENNV,PHAI,NGAYSINH,DIACHI,SODT,* *DECODE (MANV, USER, LUONG, NULL)* *LUONG,* *DECODE (MANV, USER, PHUCAP, NULL)* *PHUCAP,* *VAITRO,MANQL,PHG* *FROM GROUP08.qlnv_nhanvien;* - Thực hiện chọn mọi thuộc tính để xem và sử dụng hàm DECODE (trong ORACLE) trên 2 cột LUONG, PHUCAP để giấu đi dữ liệu LUONG, PHUCAP của nhân viên khác và chỉ hiện dữ liệu

	LUONG, PHUCAP của người dùng đang đăng nhập. - Tạo trigger để người dùng có thể thêm vào quan hệ NHANVIEN *CREATE OR REPLACE TRIGGER **NS_Them_NV*** *INSTEAD OF **INSERT*** *ON **NS_XemThongTin_NV*** *FOR EACH ROW* *BEGIN* *INSERT INTO GROUP08.QLNV_NHANVIEN(MANV,TENNV,PHAI,NGAYSINH,DIACHI,SODT,LUONG,PHUCAP,VAITRO,MANQL,PHG)* *VALUES(:NEW.MANV,:NEW.TENNV,:NEW.PHAI,:NEW.NGAYSINH,:NEW.DIACHI,:NEW.SODT,NULL,NULL,:NEW.VAITRO,:NEW.MANQL,:NEW.PHG);* *END;* - Tạo trigger để người dùng có thể cập nhật trên quan hệ nhân viên thông qua view *CREATE OR REPLACE TRIGGER **NS_CapNhatThongTin_NV*** *INSTEAD OF **UPDATE*** *ON **NS_XemThongTin_NV*** *FOR EACH ROW* *BEGIN* *IF (:OLD.LUONG IS NULL) AND (:OLD.PHUCAP IS NULL) THEN* *UPDATE GROUP08.qlnv_NHANVIEN* *SET TENNV=:NEW.TENNV, PHAI=:NEW.PHAI, NGAYSINH=:NEW.NGAYSINH, SODT=:NEW.SODT,MANQL=:NEW.MANQL, PHG=:NEW.PHG* *where manv=:new.manv;* *end if;* *END;* - Đối với trigger này, thực hiện kiểm tra nếu giá trị tại LUONG, PHUCAP đều mang giá trị mặc định là NULL thì mới thực hiện cập nhật.

	- Cuối cùng thực hiện cấp quyền xem, cập nhật và thêm trên view này cho người dùng, theo lệnh: o GRANT SELECT ON NS_XemThongTin_NV TO NHANSU o GRANT INSERT, UPDATE ON NS_XemThongTin_NV TO NHANSU

2.1.6 Chính sách 6

- Nội dung: Những người dùng có vai trò là "**Trưởng đề án**" cho biết đó là nhân viên là trưởng các đề án và có các quyền như trong bảng và tương ứng các giải pháp của nhóm

Quyền	Giải pháp
Có quyền như một nhân viên thông thường	- Thực hiện cấp các quyền tương ứng như đã cấp cho ROLE NVQL, như sau: o GRANT SELECT ON **xemthongtincanhan** to TRUONGDEAN; o GRANT UPDATE (NGAYSINH,DIACHI,SODT) ON xemthongtincanhan to TRUONGDEAN; o GRANT SELECT ON GROUP08.XemThongTinPhanCong TO TRUONGDEAN; o GRANT SELECT ON GROUP08.QLNV_PHONGBAN TO TRUONGDEAN; o GRANT SELECT ON GROUP08.QLNV_DEAN TO TRUONGDEAN;
Được quyền thêm, xóa, cập nhật trên quan hệ DEAN	- Thực hiện cấp quyền thêm, xóa, cập nhật trên quan hệ DEAN theo lệnh: GRANT INSERT, UPDATE, DELETE ON GROUP08.QLNV_DEAN TO TRUONGDEAN

2.2 Giải pháp mã hóa

- Nhóm em đã lựa chọn user ADMIN đã được tạo để thực hiện việc mã hóa
- Dữ liệu sẽ được mã hóa ở mức Database trên thuộc tính LUONG và PHUCAP do dữ liệu trên 2 thuộc tính là thuộc tính nhạy cảm. Do đó, việc mã hóa dữ liệu ở mức này sẽ giúp:
    - Bảo vệ thông tin quan trọng khỏi việc truy cập trái phép.
    - Tránh bị tấn công ở mức ứng dụng
    - Ngăn chặn các thay đổi từ người dùng không được ủy quyền
    - Ẩn được các thông tin nhạy cảm
- Nhóm không thay đổi cấu trúc dữ liệu.

- Với phương pháp mã hóa dữ liệu đã đề xuất, nhóm trình bày các khía cạnh của cơ chế quản lý khóa đề nghị:
    - Sử dụng bộ thuật toán mã hóa DBMS_CRYPTO.DES_CBC_PKCS5
    - Thiết lập khóa: Sử dụng thuật toán hash cho MANV để lấy chuỗi sau khi hash làm khóa.
    - Lưu trữ khóa: Lưu toàn bộ khóa trong bảng GROUP08_PLNV
    - Phân phối khóa: Dùng khóa đối xứng, mỗi người dùng sẽ được cấp 1 khóa để có thể giải mã
    - Phục hồi khóa khi người dùng quên khóa: Chạy lại hàm hash để xin khóa
    - Thay đổi khóa sau một thời gian: Nhóm chưa thực hiện

2.3 Audit

- Để ghi vết các hành vi, nhóm chủ yếu sử dụng FGA để cài chính sách thực hiện ghi vết, cụ thể như sau

Yêu cầu	Giải pháp
Ghi vết những hành vi của những người đã cập nhật trường THOIGIAN trong quan hệ PHANCONG	- Sử dụng DBMS_FGA để thêm chính sách trên quan hệ PHANCONG, cụ thể: *BEGIN* *DBMS_FGA.ADD_POLICY(* *object_schema => 'GROUP08'* *, object_name => 'QLNV_PHANCONG'* *, policy_name => 'FGA_UPDATE_PHANCONG_THOIGIAN'* *, audit_condition => NULL* *, audit_column => 'THOIGIAN'* *, handler_schema => NULL* *, handler_module => NULL* *, enable => TRUE* *, statement_types => 'UPDATE'* *);* *END;* - Do là thực hiện ghi vết trong mọi trường hợp, do đó để audit_condition là NULL, và cột được chỉ định ghi vết là THOIGIAN và thao tác chỉ định là cập nhật (UPDATE)
Ghi vết những hành vi của những người đã đọc trên trưởng LUONG và PHUCAP của người khác	- Sử dụng DBMS_FGA để thêm chính sách trên quan hệ NHANVIEN, cụ thể: *BEGIN* *DBMS_FGA.ADD_POLICY(* *object_schema => 'GROUP08'* *, object_name => 'QLNV_NHANVIEN'*

	, policy_name => 'FGA_NHANVIEN_READ_LUONG_PHUCAP' *, audit_condition => 'MANV != SYS_CONTEXT("USERENV", "SESSION_USER")'* *, audit_column => 'LUONG, PHUCAP'* *, handler_schema => NULL* *, handler_module => NULL* *, enable => TRUE* *, statement_types => 'SELECT'* *);* *END* - Đối với trường hợp này, nhóm đã sử dụng hàm SYS_CONTEXT để lấy ra USERNAME và tiến hành so sánh với MANV mà xuất hiện trong các dòng kết quả mà người dùng đọc được và chỉ định 2 cột là LUONG, PHUCAP. Nếu MANV không khớp với USERNAME người dùng đang đăng nhập thì thực hiện ghi vết lại do họ đang đọc giá trị LUONG, PHUCAP của người dùng khác với thao tác chỉ định là xem (SELECT)
Ghi vết hành vi của người không thuộc vai trò "Tài chính" nhưng đã cập nhật thành công trên trường LUONG và PHUCAP	- Sử dụng DBMS_FGA để thêm chính sách trên quan hệ NHANVIEN, cụ thể: begin dbms_fga.add_policy (object_schema => 'GROUP08', object_name => 'QLNV_NHANVIEN', policy_name => 'QLNV_NHANVIEN_FGA', audit_column => 'LUONG,PHUCAP', audit_condition => 'SYS_CONTEXT("SYS_SESSION_ROLES", "TAICHINH")IN ("FALSE")', statement_types => 'UPDATE', audit_column_opts => dbms_fga.any_columns); end; - Đối với trường hợp này, nhóm có điều kiện ghi vết (audit_condition) là sử dụng hàm SYS_CONTEXT với tham số phù hợp để lấy ra ROLE của người dùng đang đăng nhập và so

	sánh với 'TAICHINH' (ROLE nhóm đã tạo cho vai trò Tài chính). Nếu kết quả của hàm SYS_CONTEXT này trả về là FALSE thì có nghĩa người dùng đang đăng nhập không thuộc vai trò Tài chính và sẽ bị ghi vết nếu cập nhật thành công 2 cột đã chỉ định là LUONG, PHUCAP (do chỉ cần cập nhật thành công một trong hai trường này thì cũng sẽ bị ghi nhận, do đó tại tham số audit_column_opts, nhóm chọn DBMA_FGA.ANY_COLUMNS. (sẽ ghi vết lại bất kể cập nhật cột nào)
Kiểm tra nhật ký hệ thống	Tạo thủ tục (Procedure) để đọc nhật ký hệ thống: *CREATE OR REPLACE PROCEDURE GetAuditTrail* *AS* *c_audit SYS_REFCURSOR;* *BEGIN* *OPEN c_audit FOR SELECT dbusername, action_name, object_schema, object_name, event_timestamp, sql_text* *FROM unified_audit_trail* *ORDER BY event_timestamp ASC;* *DBMS_SQL.RETURN_RESULT(c_audit);* *END;* - Trong thủ tục, nhóm thực hiện chọn các cột như o Dbusername: Tên người dùng o Action_name: hành động thực hiện trên đối tượng o Object_schema: Schema cần được ghi vết o Object_name: tên đối tượng (ví dụ như tên bảng/view,..) o Event_timestamp: Thời gian thực hiện o Sql_text: Nội dung câu truy vấn đã thực hiện

3. TÓM LƯỢC KIẾN THỨC (Nhóm đã học được và sử dụng trong đồ án)

3.1 Cơ chế điều khiển truy cập (Access Control)

- DAC (Discretionary Access Control): Là một cơ chế điều khiển truy cập mà người dùng có thể cấp quyền cho một đối tượng/người dùng cụ thể.
- RBAC (Role-based Access Control): Là một cơ chế điều khiển truy cập mà người dùng cấp quyền cho ROLE cụ thể (một ROLE thì gồm người thể hiện người dùng khác nhau)

➔ Tùy thuộc vào trường hợp mà hệ thống sẽ thực hiện cấp quyền theo DAC hay RBAC. Nếu hệ thống có nhiều thể hiện người dùng và có nhiều quyền sử dụng chung thì nên thêm người dùng vào ROLE và thực hiện phân quyền theo cơ chế RBAC. Ngược lại, nếu không có quá nhiều thể hiện người dùng thì có thể phân quyền cho từng người dùng cụ thể.

- Bên cạnh đó, nhóm còn thấy được rằng việc tạo VIEW giúp ích rất nhiều trong việc phân quyền cho người dùng thao tác trên dữ liệu. Cụ thể, view giúp bảo vệ được bảng gốc của cơ sở dữ liệu, đồng thời vẫn cấp được các quyền thao tác cần thiết cho người dùng. VIEW còn giúp giới hạn quyền trên dòng, và nếu sử dụng thêm hàm DECODE được hỗ trợ bởi Hệ quản trị ORACLE thì người dùng còn có thể thực hiện việc giấu đi một số dữ liệu nhạy cảm trên cột đó.
- Để một VIEW có thể chỉnh sửa được và thêm được thì VIEW đó phải được cài thêm trigger cho các thao tác tương ứng. (Sử dụng INSTEAD OF TRIGGER). Việc cài trigger này sẽ giúp người dùng có thể chỉnh sửa, thêm, xóa trên các VIEW phức tạp như các VIEW gồm nhiều bảng, có sử dụng các hàm có sẵn,..

3.2 Audit

- Việc AUDIT là thực hiện ghi vết lại những thao tác mà người dùng đã thực hiện. Để thực hiện việc AUDIT, nhóm đã sử dụng package có sẵn trong Hệ quản trị ORACLE để cài các chính sách theo yêu cầu đề bài và thực hiện AUDIT. Cụ thể, gói DBMS_FGA cung cấp chức năng bảo mật chi tiết (Fine-grained Auditing). Việc sử dụng FGA cho phép:
    - Áp dụng đối với mọi mệnh đề thao tác trên dữ liệu (DML – Data Manipulation Language)
    - Có thể được mở rộng đến standard audit
    - Có thể tinh chỉnh việc kích hoạt ghi vết khi người dùng thực hiện các thao tác trên các cột chỉ định và thỏa điều kiện cho trước.
    - FGA có các điểm mạnh hơn so với phương pháp ghi vết chuẩn (standard audit) như : Có điều kiện kiểm tra (Boolean), chỉ định trên các cột có dữ liệu nhạy cảm,.
- Đối với thủ tục Thêm chính sách (ADD-POLICY), nếu để điều kiện là NULL thì sẽ luôn thực hiện ghi vết với các trường hợp thỏa chính sách đó.
- Các thủ tục mà gói DBMS_FGA hỗ trợ :
    - ADD_POLICY : Được dùng để thêm chính sách ghi vết cho một đối tượng
    - DISABLE_POLICY : Được dùng để vô hiệu hóa chính sách
    - DROP_POLICY : Được dùng để bỏ/xóa chính sách
    - ENABLE_POLICY : Được dùng để kích hoạt chính sách
- Cú pháp thêm chính sách :

```
DBMS_FGA.ADD_POLICY(
  object_schema      IN  VARCHAR2 DEFAULT NULL,
  object_name        IN  VARCHAR2,
```

```
policy_name        IN  VARCHAR2,
audit_condition    IN  VARCHAR2 DEFAULT NULL,
audit_column       IN  VARCHAR2 DEFAULT NULL,
handler_schema     IN  VARCHAR2 DEFAULT NULL,
handler_module     IN  VARCHAR2 DEFAULT NULL,
enable             IN  BOOLEAN DEFAULT TRUE,
statement_types    IN  VARCHAR2 DEFAULT SELECT,
audit_trail        IN  BINARY_INTEGER DEFAULT NULL,
audit_column_opts  IN  BINARY_INTEGER DEFAULT ANY_COLUMNS,
policy_owner       IN  VARCHAR2 DEFAULT NULL);
```

- Tham số và ý nghĩa tham số

Tham số	Ý nghĩa
Object_schema	Tên schema của đối tượng bị ghi vết. Nếu tham số này mang giá trị NULL thì schema mặc định sẽ là schema hiện hành.
Object_name	Tên của đối tượng bị ghi vết
Policy_name	Tên chính sách (phải khác nhau) và không được thêm các ký tự đặc biệt vào.
Audit_condition	Điều kiện của việc kích hoạt chính sách. Nếu để trống (NULL) thì giá trị này được xem là TRUE và sẽ kích hoạt chính sách. Điều kiện phải đơn giản và không được quá phức tạp
Audit_column	Cột được chỉ định để kiểm tra cho việc bị truy cập. Nếu để NULL (mặc định) thì sẽ thực hiện ghi vết nếu có bất kỳ cột nào của đối tượng bị truy xuất hoặc thay đổi.
Handler_schema	Schema mà lưu trữ xử lý sự kiện. Nếu để NULL thì mặc định là schema hiện hành
Handler_module	Tên của hàm xử lý sự kiện/
Enable	Kích hoạt chính sách nếu giá trị tham số là TRUE (và giá trị mặc định là TRUE)

Statement_types	Loại thao tác được ghi vết, gồm: INSERT, UPDATE, DELETE, SELECT
Audit_trail	Nếu người dùng đã chuyển sang chế độ ***unified auditing*** thì nên bỏ qua tham số này vì mọi bản ghi sẽ được lưu vào trong bảng **unified audit trail**. Và nếu người dùng muốn sử dụng thì tham số này chỉ định nơi viết bản ghi ghi vết và xem xét rằng có bao gồm thông tin của biến SQL Text và SQL bind.. Ví dụ nếu người dùng chỉnh giá trị của tham số ***audit_trail thành DBMS_FGA.DB:*** Thì các bản ghi sẽ được lưu vào bảng SYS.FGA_LOG$ và bỏ qua SQL Text và SQL Bind
Audit_column_opts	Lựa chọn kích hoạt chính sách khi người dùng thao tác trên mọi cột (all_columns) hoặc chỉ cần thao tác trên bất kỳ cột nào (any_columns)

III. TÀI LIỆU THAM KHẢO

https://drive.google.com/drive/folders/1gaMJU3OSKZxDH4Nxc4a6Lt-jzaYAMKey

https://docs.oracle.com/en/database/oracle/oracle-database/21/arpls/DBMS_FGA.html