

# CHƯƠNG 2: ĐIỀU KHIỂN TRUY CẬP

# Nội dung

- 2.1. Tổng quan về điều khiển truy cập
- 2.2. Điều khiển truy cập tùy ý
- 2.3. Điều khiển truy cập bắt buộc
- 2.4. Điều khiển truy cập dựa trên vai trò

## 2.1. Tổng quan về điều khiển truy cập

- Về cơ bản, bảo mật CSDL dựa trên việc điều khiển truy cập vào hệ thống CSDL. Điều khiển truy cập vật lý là một phần của bảo mật CSDL. Phần chính khác là điều khiển truy cập thông qua DBMS (*DataBase Management System - hệ quản trị CSDL*).
- Điều khiển truy cập nhằm bảo đảm cho các truy cập trực tiếp đến đối tượng được ủy quyền

## 2.1. Tổng quan về điều khiển truy cập

- **Các đối tượng liên quan đến điều khiển truy cập:**
  - *Đối tượng (Objects)* cần bảo vệ: các tài nguyên phần cứng, phần mềm, dữ liệu của hệ thống (tệp, bảng dữ liệu, các bộ giá trị, ...)
  - *Chủ thể (Subjects)* yêu cầu truy cập: các cá nhân/đơn vị/chương trình yêu cầu truy cập đến tài nguyên (VD: chủ sở hữu, người dùng, chương trình ứng dụng, ...). *Mức truy cập dữ liệu (Level of data access)* với mỗi chủ thể/loại chủ thể là khác nhau (quyền truy cập đến các đối tượng CSDL ở các mức chi tiết khác nhau).
  - *Chế độ truy cập (Access modes)*: kiểu truy cập ứng với các thao tác tạo, đọc, chèn, sửa, xóa, di chuyển, thực thi, xác nhận sự tồn tại

**Subjects → Access request → Reference monitor → Object**

## 2.1. Tổng quan về điều khiển truy cập

*Mức truy cập dữ liệu:*

- Một phương pháp thực thi bảo mật tổng quát phải có phạm vi lớn và tính linh hoạt lớn.
- Một hệ thống bảo mật linh hoạt trong DBMS phải có khả năng cấp các đặc quyền ở các mức dữ liệu sau:
  - + Toàn bộ CSDL
  - + Quan hệ riêng lẻ:
    - Tất cả các hàng và tất cả các cột
    - Tất cả các hàng nhưng chỉ các cột cụ thể
    - Tất cả các cột nhưng chỉ các hàng cụ thể
    - Các hàng và các cột cụ thể

## 2.1. Tổng quan về điều khiển truy cập

*Mức truy cập dữ liệu (tiếp):*

- Ví dụ: Một CSDL quan hệ của một công ty xây dựng có chứa quan hệ WORKER (Công nhân):

*WORKER (WorkerId, Name, Address, City, State, Zip, SuperId, WageRate)*

→ Các mức đặc quyền truy cập dữ liệu có thể cấp cho người dùng:

- + Người dùng toàn quyền truy cập (không bị giới hạn) đối với toàn bộ quan hệ *WORKER*
- + Người dùng không có quyền truy cập dưới bất kỳ hình thức nào đối với bất kỳ phần nào của quan hệ *WORKER*.
- + Người dùng chỉ có thể đọc nhưng không được phép thay đổi bất kỳ dữ liệu nào của quan hệ *WORKER*
- + Người dùng có thể đọc dòng (bản ghi) chứa dữ liệu của mình nhưng không thể thay đổi bất kỳ cột (trường) nào trong dòng đó.
- + Người dùng chỉ có thể đọc dòng của mình trong mỗi quan hệ nhưng chỉ có thể thay đổi các cột *Name, Address*.
- + Người dùng có thể đọc các cột *WorkerId, Name, Address, SuperId* của bất kỳ dòng nào nhưng chỉ có thể thay đổi *Name, Address*.

## 2.1. Tổng quan về điều khiển truy cập

*Chế độ truy cập dữ liệu:*

- *Tạo (Create) hoặc Chèn (Insert):* Thêm dữ liệu vào một tệp mà không hủy bất kỳ dữ liệu nào.
- *Đọc (Read):* Người dùng có thể đọc và sao chép dữ liệu từ CSDL vào môi trường người dùng thông qua các chương trình ứng dụng hoặc các truy vấn CSDL.
- *Cập nhật (Update):* Ghi các giá trị cập nhật.
- *Xóa (Delete):* Xóa/hủy các đối tượng dữ liệu cụ thể.
- *Di chuyển (Move):* Di chuyển các đối tượng dữ liệu mà không có đặc quyền đọc nội dung.
- *Thực thi (Execute):* Chạy một chương trình hoặc thủ tục với các đặc quyền cần thiết cho việc thực thi.
- *Xác nhận sự tồn tại (Verify Existence):* Xác minh xem một đối tượng CSDL cụ thể có tồn tại trong CSDL hay không.

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Chức năng của điều khiển truy cập:*

- Cấp phép hoặc từ chối phê duyệt yêu cầu truy cập/sử dụng các tài nguyên (đối tượng) xác định cho các chủ thể.
- Kiểm soát được các đối tượng đang hoạt động hay các đối tượng có thể bị truy cập bởi các hoạt động khác.

### ▪ *Các thành phần điều khiển truy cập:*


- Chính sách điều khiển truy cập: quy định cụ thể thẩm quyền truy cập của một hệ thống
- Cơ chế điều khiển truy cập: thực thi các chính sách điều khiển truy cập




## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các bước điều khiển truy cập:*

**Định danh (Identification):** Người dùng cung cấp định danh (ví dụ: username); hệ thống nhận diện định danh do người dùng cung cấp



**Xác thực (Authentication):** Người dùng chứng minh định danh đó là đúng, hợp lệ (ví dụ: cung cấp password); hệ thống xác minh, công nhận định danh của người dùng



**Ủy quyền (Authorization):** Hệ thống xác định, cấp quyền truy cập cho người dùng (ví dụ: quyền đọc, chèn, sửa, xóa các tệp dữ liệu)

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các vai trò (role) trong điều khiển truy cập:*

- *Chủ sở hữu:* Người sở hữu, chịu trách nhiệm về dữ liệu, chịu trách nhiệm xác định mức bảo mật cần thiết đối với dữ liệu và giao phó các nhiệm vụ bảo mật khi cần
- *Người giám sát:* Cá nhân thực hiện các hành động giám sát do chủ sở hữu giao phó, chịu trách nhiệm thường xuyên rà soát các thiết lập bảo mật và duy trì truy cập của người dùng
- *Người dùng:* Người truy cập dữ liệu trong quyền hạn được giao, chịu trách nhiệm tuân thủ đúng các chỉ dẫn, các yêu cầu bảo mật của tổ chức, không được vi phạm bảo mật

## 2.1. Tổng quan về điều khiển truy cập

- *Ai có thẩm quyền cấp và thu hồi quyền truy cập?*
  - Centralized administration (quản trị tập trung): nhân viên an ninh
  - Decentralized administration (quản trị phân cấp): hệ thống phân cấp theo quản lý
  - Hierarchical decentralization (phân cấp theo thứ bậc): nhân viên an ninh → quản trị hệ thống phòng ban → người quản trị Windows
  - Ownership based (quyền sở hữu): có thể cấp quyền truy cập cho những người khác truy xuất dữ liệu của mình
  - Cooperative authorization (ủy quyền): ủy quyền cho các chủ thể khác

## 2.1. Tổng quan về điều khiển truy cập

- *Những khó khăn trong điều khiển truy cập?*
  - Lượng yêu cầu truy cập hệ thống lớn, thường xuyên
  - Các yêu cầu điều khiển truy cập có thể thay đổi
  - Phải tuân theo các ràng buộc của hệ thống
  - Vấn đề an toàn

## 2.1. Tổng quan về điều khiển truy cập

- *Các điều kiện điều khiển truy cập:*
  - Điều khiển truy cập dựa theo tên truy cập
  - Điều khiển truy cập dựa trên dữ liệu
  - Điều khiển truy cập dựa trên thời gian
  - Điều khiển truy cập dựa trên ngữ cảnh
  - Điều khiển truy cập dựa vào lịch sử

## 2.1. Tổng quan về điều khiển truy cập

### ▪ Các điều kiện điều khiển truy cập (tiếp):

- Điều khiển truy cập dựa theo tên truy cập (*Access Controlled by Name*): hạn chế truy cập dựa trên tên truy cập

*Ví dụ:*

<b>Chủ thể \ Đối tượng</b>	<b>Tập F1</b>	<b>Tập F2</b>	<b>Tập F3</b>
Người dùng 1	R, W	R, W	Exec
Người dùng 2	R, W	R	-
Chương trình 1	R	R, W	-
Chương trình 2	R	-	Del

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các điều kiện điều khiển truy cập (tiếp):*

- *Điều khiển truy cập dựa trên dữ liệu (Access Controlled by Data):* hạn chế truy cập dựa trên dữ liệu.

*Ví dụ:* Chỉ được xem thông tin của các Worker có *City = 'NewYork'* trong bảng *WORKER*.

- *Điều khiển truy cập dựa trên thời gian (Access Controlled by Time):* hạn chế truy cập dựa trên thời gian truy cập.

*Ví dụ:* Chỉ được truy cập bảng *WORKER* trong khoảng thời gian từ 8:00 đến 18:00 từ *Monday* đến *Friday* hàng tuần.

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các điều kiện điều khiển truy cập (tiếp):*

- *Điều khiển truy cập dựa trên ngữ cảnh (Access Controlled by Context):* hạn chế truy cập dựa trên bối cảnh thu thập thông tin có thể truy cập.

*Ví dụ:* có thể truy cập riêng từng thuộc tính *WorkerId*, *Name*, *Address* trong bảng *WORKER*, nhưng không thể truy cập cùng lúc cả 3 thuộc tính này.

- *Điều khiển truy cập dựa trên lịch sử (Access Controlled by History):* hạn chế truy cập dựa trên dữ liệu truy cập trước.

*Ví dụ:* Chỉ truy cập được vào thuộc tính *Address* nếu trước đó chưa truy cập vào thuộc tính *Name*.



## 2.1. Tổng quan về điều khiển truy cập

- ***Thực thi điều khiển truy cập:***
  - Danh sách điều khiển truy cập
  - Chính sách nhóm
  - Giới hạn tài khoản

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Thực thi điều khiển truy cập (tiếp):*

- *Danh sách điều khiển truy cập (Access Control List-ACL):*
  - + Tập các quyền gắn với một đối tượng
  - + Xác định chủ thể nào có thể truy cập tới đối tượng và các thao tác nào mà chủ thể thực hiện
  - + Khi chủ thể yêu cầu thực hiện một thao tác trên đối tượng: Hệ thống kiểm tra danh sách điều khiển truy cập với mục đã được phê duyệt
  - + Danh sách điều khiển truy cập thường được xem xét trong mối liên hệ với các file của hệ điều hành

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Thực thi điều khiển truy cập (tiếp):*

- *Danh sách điều khiển truy cập (Access Control List-ACL) (tiếp):*

Mỗi một mục trong bảng danh sách điều khiển truy cập được gọi là một mục điều khiển (Access Control Element - ACE). Cấu trúc ACE trong Windows:

- + Nhận dạng truy cập (Access identifier): cho tài khoản người dùng hay tài khoản nhóm hoặc phiên đăng nhập
- + Mặt nạ truy cập (Access mask) xác định quyền truy cập do ACE điều khiển
- + Cờ (Flag): cho biết kiểu của ACE
- + Tập các cờ (Set of flags): xác định đối tượng có thể kế thừa các quyền hay không

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Thực thi điều khiển truy cập (tiếp):*

#### - *Chính sách nhóm (Group Policy):*

##### + Tính năng của Microsoft Windows:

- Cho phép sử dụng Active Directory (AD) để quản lý và cấu hình tập trung cho các máy và người dùng từ xa.
- Thường được sử dụng trong các môi trường doanh nghiệp
- Các thiết lập được lưu trữ trong các GPO (Group Policy Object - chính sách cho nhóm đối tượng)

##### + Local Group Policy:

- Có ít tùy chọn hơn so với Group Policy
- Được sử dụng để cấu hình các thiết lập cho các hệ thống không phải là một phần của AD

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Thực thi điều khiển truy cập (tiếp):*

#### - *Giới hạn tài khoản:*

+ Giới hạn thời gian trong ngày (time of day restriction)

- Giới hạn số lần người dùng đăng nhập vào hệ thống trong một ngày
- Cho phép chọn khối thời gian chặn các truy cập
- Có thể được thiết lập trên từng hệ thống riêng lẻ

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Thực thi điều khiển truy cập (tiếp):*

#### - *Giới hạn tài khoản (tiếp):*

##### + Hạn sử dụng tài khoản (account expiration):

- Các tài khoản “mồ côi” (orphaned account): tài khoản vẫn hoạt động sau khi chủ thể đã rời khỏi tổ chức
- Tài khoản “ngủ đông” (domant account): không hoạt động trong một khoảng thời gian dài

→ Cả hai kiểu tài khoản “mồ côi” và “ngủ đông” đều là những nguy cơ đối với bảo mật. Các khuyến cáo xử lý:

- Thiết lập một quy trình chính thức
- Chấm dứt truy cập ngay lập tức
- Quản lý nhật ký (file log)
- Thiết lập hết hạn cho một tài khoản người dùng (hết hiệu lực)

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Thực thi điều khiển truy cập (tiếp):*

#### - *Giới hạn tài khoản (tiếp):*

#### + Hạn sử dụng tài khoản (account expiration):

- Password expiration (thời gian hiệu lực của mật khẩu): thiết lập khoảng thời gian mà người dùng phải thay đổi mật khẩu mới
- Account expiration (thời gian hiệu lực của tài khoản) có thể được thiết lập bằng số ngày mà người dùng không có bất cứ hành động truy cập nào

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các dịch vụ xác thực:*

- *Xác thực (Authentication):* Quá trình xác minh thông tin
- Dịch vụ xác thực được cung cấp trên một mạng: Máy chủ xác thực chuyên dụng (còn gọi là máy chủ AAA nếu nó thực hiện đồng thời cả nhiệm vụ ủy quyền (authorization) và kiểm toán (accounting))
- Các kiểu xác thực và máy chủ AAA thông dụng:
  - + RADIUS
  - + KERBEROS
  - + TACACS
  - + LDAP



## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các dịch vụ xác thực (tiếp):*

- *RADIUS (Remote Authentication Dial In User Service - Bộ quay số xác thực từ xa trong dịch vụ người dùng):*
  - + Được giới thiệu vào năm 1992, sau đó đã trở thành một tiêu chuẩn công nghiệp và vẫn được sử dụng cho đến nay
  - + Phù hợp cho các ứng dụng kiểm soát dịch vụ cỡ lớn, ví dụ như truy cập quay số tới mạng doanh nghiệp
  - + *RADIUS client:*
    - Thường là một thiết bị như điểm truy cập không dây (AP)
    - Có nhiệm vụ gửi các thông tin về người dùng cùng với các tham số kết nối tới máy chủ RADIUS

## 2.1. Tổng quan về điều khiển truy cập

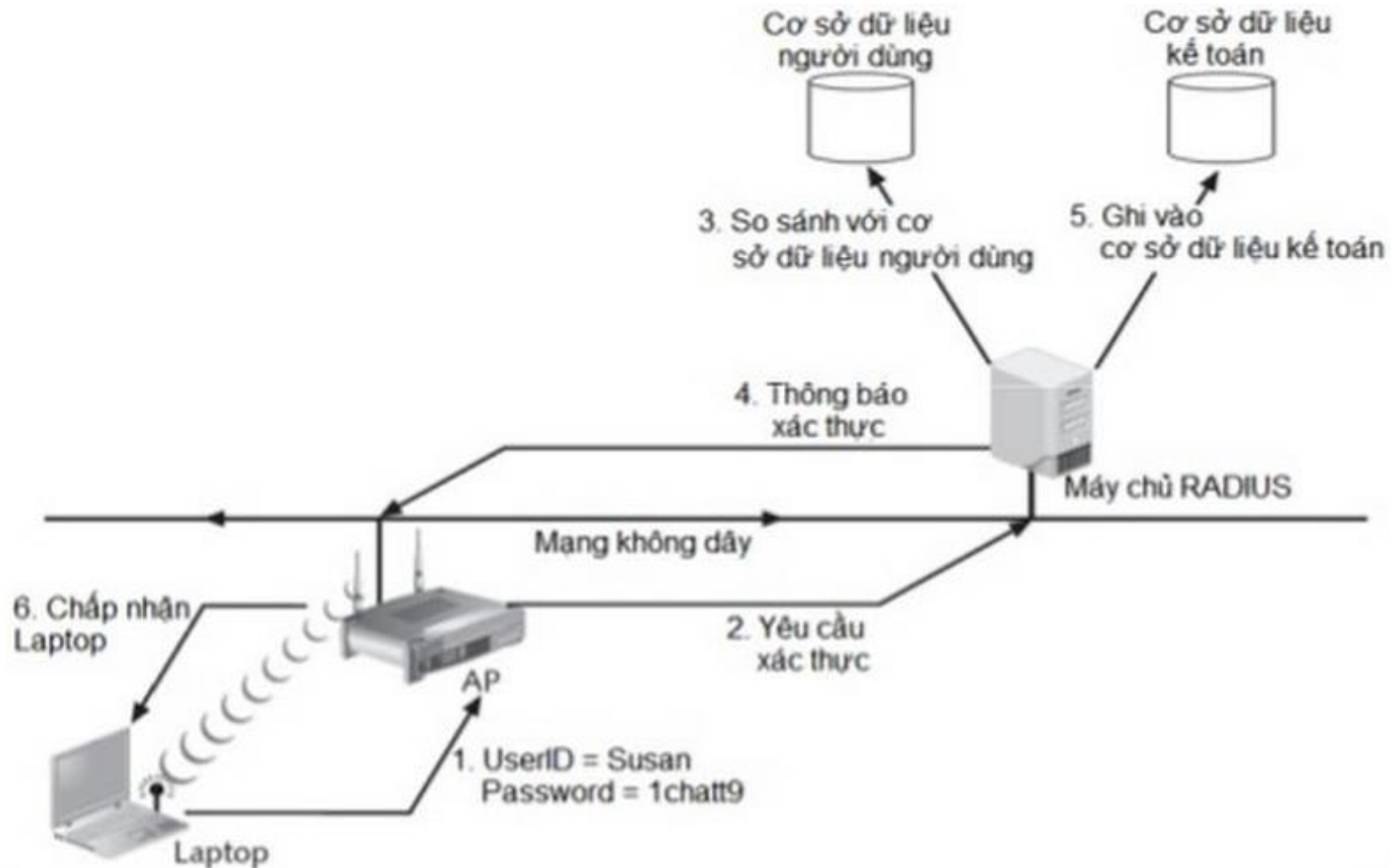
### ▪ *Các dịch vụ xác thực (tiếp):*

#### - *RADIUS (tiếp):*

- + Hồ sơ người dùng RADIUS được lưu trữ trong CSDL trung tâm, tất cả các máy chủ từ xa đều có thể chia sẻ thông tin
- + Ưu điểm của dịch vụ trung tâm:
  - Tăng cường bảo mật do chỉ có duy nhất một điểm quản lý trên mạng
  - Dễ dàng theo dõi và truy vết việc sử dụng để thanh toán và lưu giữ các số liệu thống kê mạng

## 2.1. Tổng quan về điều khiển truy cập

- *Các dịch vụ xác thực (tiếp):*
  - + *Xác thực RADIUS:*



## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các dịch vụ xác thực (tiếp):*

#### - *KERBEROS:*

- + Hệ thống xác thực được phát triển tại MIT, sử dụng mã hóa và xác thực để đảm bảo tính bảo mật
- + Thường được sử dụng cài đặt trong các thiết lập giáo dục và chính phủ
- + Hoạt động giống như việc sử dụng giấy phép lái xe để thanh toán SEC
- + *Vé Kerberos:*
  - Chứa các thông tin liên quan tới người dùng
  - Người dùng trình diện vé vào mạng cho một dịch vụ
  - Rất khó để sao chép
  - Hết hiệu lực sau một vài giờ hoặc sau một ngày

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các dịch vụ xác thực (tiếp):*

- *TACACS (Terminal Access Control Access Control System - Hệ thống điều khiển truy cập điều khiển truy cập thiết bị đầu cuối)*
  - + Dịch vụ xác thực tương tự như RADIUS do Cisco System phát triển
  - + Thường được sử dụng trên các thiết bị UNIX
  - + Giao tiếp bằng cách chuyển tiếp thông tin xác thực người dùng tới một máy chủ trung tâm
  - + Phiên bản hiện tại là TACACS+

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các dịch vụ xác thực (tiếp):*

- *LDAP (Lightweight Directory Access Control System - Giao thức truy cập thư mục hạng nhẹ)*

+ Dịch vụ thư mục

- CSDL được lưu trên mạng
- Chứa các thông tin về người dùng và các thiết bị mạng
- Lưu vết theo dõi các tài nguyên mạng và đặc quyền của người dùng đối với những tài nguyên đó
- Cho phép hay từ chối truy cập dựa trên thông tin lưu trữ

+ Tiêu chuẩn cho các dịch vụ thư mục: X.500

+ DAP (Directory Access Protocol - Giao thức truy cập thư mục)

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Các dịch vụ xác thực (tiếp):*

#### - *LDAP (tiếp)*

+LDAP: Là một tập con đơn giản hơn của DAP

- Được thiết kế để hoạt động trên bộ giao thức TCP/IP
- Có các chức năng đơn giản hơn
- Mã hóa các thành phần giao thức theo cách đơn giản hơn so với X.500
- Là một giao thức mở

+Nhược điểm của LDAP: Có thể là mục tiêu của tấn công lan truyền LDAP:

- Tương tự như tấn công lan truyền SQL
- Xảy ra khi dữ liệu do người dùng cung cấp không được lọc đúng cách

## 2.1. Tổng quan về điều khiển truy cập

- *Các mô hình điều khiển truy cập (Access Control Models):*
  - *4 mô hình chính:*
    - + Điều khiển truy cập tùy ý
    - + Điều khiển truy cập bắt buộc
    - + Điều khiển truy cập dựa trên vai trò
    - + Điều khiển truy cập dựa trên quy tắc



## 2.1. Tổng quan về điều khiển truy cập

- **Các mô hình điều khiển truy cập (Access Control Models):**
  - **Điều khiển truy cập tùy ý (DAC - Discretionary Access Control):**
    - + Cho biết chủ thể nào có thể truy cập kiểu gì đến các đối tượng trong CSDL
    - + Có những nguyên tắc một chủ thể có thể tùy ý cấp quyền hay lấy lại quyền truy cập hoặc truy cập gián tiếp đến lớp dữ liệu
  - **Điều khiển truy cập bắt buộc (MAC - Mandatory Access Control):** Định trước các nguyên tắc để chủ thể (thuộc 1 lớp) truy cập trực tiếp hoặc gián tiếp đến các lớp dữ liệu
  - **Điều khiển truy cập dựa trên vai trò (RBAC - Role Based Access Control):** Vai trò là 1 tập các quyền, RBAC không thực hiện cấp quyền cho từng chủ thể mà gán cho chủ thể 1 vai trò, khi đó chủ thể sẽ có tất cả các quyền được cấp cho vai trò đó

## 2.1. Tổng quan về điều khiển truy cập

- *Các mô hình điều khiển truy cập (Access Control Models):*
  - *Điều khiển truy cập dựa trên qui tắc (Rule Based Access Control):*
    - + Tự động gán vai trò cho các chủ thể dựa trên một tập quy tắc do người giám sát xác định
    - + Mỗi đối tượng tài nguyên chứa các thuộc tính truy cập dựa trên quy tắc
    - + Khi người dùng truy cập tới tài nguyên, hệ thống sẽ kiểm tra các qui tắc của đối tượng để xác định quyền truy cập
    - + Thường được sử dụng để quản lý truy cập người dùng tới một hoặc nhiều hệ thống
    - + Những thay đổi trong doanh nghiệp có thể làm cho việc áp dụng các qui tắc thay đổi

## 2.1. Tổng quan về điều khiển truy cập

### ▪ Các mô hình điều khiển truy cập (*Access Control Models*):

Tên mô hình	Mô tả	Hạn chế
Điều khiển truy cập tùy ý	Là mô hình nghiêm ngặt nhất	Người dùng không thể thiết lập điều khiển
Điều khiển truy cập bắt buộc	Là mô hình cởi mở nhất	Chủ thể có toàn quyền với đối tượng
Điều khiển truy cập dựa trên vai trò	Là phương pháp thực tế hơn	Cấp quyền cho các vai trò cụ thể trong tổ chức, sau đó chỉ định vai trò cho người dùng
Điều khiển truy cập dựa trên quy tắc	Được sử dụng để quản lý truy cập người dùng tới một hoặc nhiều hệ thống	Tự động gán các vai trò cho các chủ thể dựa trên tập quy tắc do người giám sát quy định

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Gợi ý thực thi điều khiển truy cập:*

- Thiết lập các thủ tục tối ưu để hạn chế truy cập, nhằm đảm bảo an toàn cho hệ thống và dữ liệu
- Một số biện pháp tối ưu:
  - + Tách nhiệm vụ (separation of duties)
  - + Luân chuyển công việc (job rotation)
  - + Đặc quyền tối thiểu (least privilege)
  - + Từ chối ngầm định (implicit deny)
  - + Kỳ nghỉ bắt buộc (mandatory vacation)

## 2.1. Tổng quan về điều khiển truy cập

- *Gợi ý thực thi điều khiển truy cập (tiếp):*

- *Tách nhiệm vụ:*

- + Hành vi gian lận có thể bắt nguồn từ việc tin cậy vào một cá nhân và cho phép họ toàn quyền điều khiển một quá trình

- Yêu cầu phải có ít nhất hai người chịu trách nhiệm cho các hoạt động liên quan tới quản lý, giúp hệ thống không bị xâm hại do hành vi của một cá nhân đơn lẻ

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Gợi ý thực thi điều khiển truy cập (tiếp):*

#### - *Luân chuyển công việc:*

##### + Luân chuyển công việc

- Luân chuyển trách nhiệm công việc của các cá nhân theo định kỳ
- Các nhân viên có thể được chuyển công việc ngay trong nội bộ một phòng ban hoặc giữa các phòng ban với nhau

##### + Ưu điểm của biện pháp Luân chuyển công việc:

- Hạn chế thời gian tại vị của các cá nhân để họ không thể thao túng các cấu hình bảo mật
- Giúp vạch trần các con đường tiềm ẩn dẫn đến gian lận
- Mỗi cá nhân có một quan điểm khác nhau và điều đó có thể giúp phát hiện ra các lỗ hổng bảo mật
- Giảm bớt căng thẳng mệt mỏi cho nhân viên

## 2.1. Tổng quan về điều khiển truy cập

- *Gợi ý thực thi điều khiển truy cập (tiếp):*

- *Đặc quyền tối thiểu:*

- + Giới hạn truy cập tới thông tin dựa trên nguyên tắc chỉ được biết những gì phục vụ cho công việc
    - + Giúp giảm thiểu tấn công thông qua việc loại bỏ các đặc quyền không cần thiết
    - + Nên áp dụng cho người dùng và tiến trình trên hệ thống
    - + Các tiến trình nên hoạt động ở cấp độ bảo mật tối thiểu cần thiết để hoạt động chính xác
    - + Khả năng gán các mức ưu tiên cao hơn cũng rất lớn

## 2.1. Tổng quan về điều khiển truy cập

### ▪ *Gợi ý thực thi điều khiển truy cập (tiếp):*

#### - *Đặc quyền tối thiểu (tiếp):*

+ Các thách thức đối với biện pháp Đặc quyền tối thiểu

- Các ứng dụng kế thừa: nhiều ứng dụng trong tổ chức không còn được bảo trì hoặc không còn được hỗ trợ bởi bên thứ 3, xây dựng lại các ứng dụng sẽ cần chi phí lớn → khắc phục: chạy các ứng dụng đó trên môi trường ảo
- Các nhiệm vụ quản trị chung: các nhiệm vụ quản trị cơ bản được thực hiện bởi người dùng, nếu không có đặc quyền cao, người dùng sẽ không thực hiện được → cần liên hệ với phụ trách kỹ thuật (bất tiện)
- Yêu cầu cập nhật/nâng cấp phần mềm: việc cập nhật/nâng cấp các phần mềm không được triển khai tập trung thường đòi hỏi đặc quyền cao → cần hỗ trợ từ phụ trách kỹ thuật → giảm năng suất và tăng chi phí hỗ trợ



## 2.1. Tổng quan về điều khiển truy cập

- *Gợi ý thực thi điều khiển truy cập (tiếp):*

- *Từ chối ngầm:*

- + Nếu một điều kiện không đáp ứng rõ ràng, yêu cầu truy cập sẽ bị từ chối
    - + Ví dụ: bộ định tuyến mạnh sẽ từ chối các truy cập trừ khi đáp ứng được các điều kiện truy cập phù hợp với các quy tắc giới hạn

## 2.1. Tổng quan về điều khiển truy cập

- *Gợi ý thực thi điều khiển truy cập (tiếp):*

- *Kỳ nghỉ bắt buộc:*

- + Thông thường, thủ phạm phải có mặt hàng ngày ở tổ chức để che dấu hành vi gian lận của mình

- Có thể lên kế hoạch kiểm tra hành vi của nhân viên giữ chức vụ nhạy cảm trong suốt thời gian nghỉ để hạn chế gian lận

## 2.2. Điều khiển truy cập tùy ý

- **Điều khiển truy cập tùy ý (DAC - Discretionary Access Control):**
  - DAC chỉ rõ những đặc quyền mà mỗi chủ thể có thể có được trên các đối tượng và trên hệ thống (object privilege, system privilege):
    - + DAC dựa vào định danh của người dùng có yêu cầu truy cập vào các đối tượng dữ liệu.
    - + Các yêu cầu truy cập được kiểm tra thông qua một cơ chế kiểm soát tùy ý, quyền truy cập sẽ được trao cho các chủ thể thỏa mãn các quy tắc trao quyền của hệ thống.
    - + Việc phân quyền kiểm soát dựa vào quyền sở hữu (kiểu chính sách cấp quyền dựa vào quyền sở hữu)

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển truy cập tùy ý (tiếp):

- Cơ chế này được gọi là tùy ý có nghĩa là:
  - + Cho phép chủ thể có thể cấp quyền cho chủ thể khác truy cập tới các đối tượng của nó
  - + Người sử dụng có khả năng cấp phát hoặc thu hồi quyền truy cập trên một số đối tượng.
- Ưu điểm:
  - + Với DAC, người dùng có thể bảo vệ những gì thuộc về mình, chủ sở hữu dữ liệu có toàn quyền trên dữ liệu và có quyền định ra quyền truy cập với các chế độ đọc/ghi/thực thi và cấp những quyền đó cho những người khác
  - + Tính linh hoạt cao
  - DAC được triển khai rộng rãi trong hầu hết các hệ điều hành

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển truy cập tùy ý (tiếp):

- *Nhược điểm của DAC*: DAC cho phép đọc thông tin từ một đối tượng này và chuyển đến (ghi vào) một đối tượng khác  
→ Tạo ra sơ hở để Trojan tấn công, sao chép thông tin.

*Ví dụ:*

- + *UserA* là chủ sở hữu *TableA*, anh ta tạo ra khung nhìn *ViewA* từ bảng này (sao chép thông tin).
- + *UserA* không cho phép *UserB* quyền đọc *TableA* nhưng lại vô tình cấp quyền *read* cho *UserB* trên *ViewA*  
→ *UserB* có thể đọc *TableA* dù không đủ quyền trên bảng này.

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Điều khiển truy cập tùy ý (tiếp):*

- *Các mô hình DAC thường có các đặc điểm:*

- + Người sở hữu dữ liệu có thể cấp quyền sở hữu /xác định chế độ truy xuất để cấp cho những người dùng khác (read, write, ...)
- + Hệ thống sẽ cảnh báo hoặc giới hạn truy xuất của người dùng trong trường hợp yêu cầu truy xuất tới tài nguyên hoặc đối tượng không đáp ứng quá trình xác thực (sau một số lần)
- + Một tiện ích tăng cường (add-on) hoặc bổ sung (plug-in) được áp dụng cho một máy khách để ngăn ngừa người dùng sao chép thông tin
- + Người dùng không có quyền truy xuất thông tin không xác định (không xác định được kích thước, tên, đường dẫn của file, ...)
- + Việc truy xuất tới thông tin được xác định dựa trên quyền truy cập hợp pháp được mô tả trong danh sách điều khiển truy cập theo danh tính người dùng và nhóm

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Các thao tác cơ bản trong DAC:*

- *Các hình thức điều khiển truy cập cơ bản của DAC trong một hệ CSDL dựa vào 2 thao tác cơ bản:*
  - + *Cấp quyền (granting privileges):* cho phép người dùng khác được quyền truy cập lên đối tượng do mình làm chủ.
  - + *Thu hồi quyền (revoking privileges):* thu hồi lại quyền đã cấp cho người dùng khác.

*Ví dụ:* Nếu *UserA* được phép cấp quyền (WITH GRANT OPTION), *UserA* có thể cấp/thu hồi quyền đã cấp cho người dùng khác.

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Quy tắc trao quyền trong DAC:*

- *Quy tắc trao quyền:* Các yêu cầu và chính sách an toàn do tổ chức đưa ra, người trao quyền có nhiệm vụ chuyển các yêu cầu này thành các quy tắc trao quyền
- Có hai cấp độ cơ bản để cấp hoặc thu hồi các đặc quyền truy cập:
  - + Người dùng (Users): Một người dùng hoặc một nhóm người dùng có thể nhận dạng
  - + Đối tượng CSDL (Database Objects): Mục/thành phần dữ liệu, thường là bảng hoặc khung nhìn (table/view)



## 2.2. Điều khiển truy cập tùy ý

### ▪ Quy tắc trao quyền trong DAC (tiếp):

- *Quyền ở cấp người dùng (còn gọi là cấp tài khoản/hệ thống - account/system level):* là những quyền độc lập với các đối tượng trong hệ CSDL, do người quản trị hệ thống định nghĩa và cấp cho mỗi người dùng. Có thể phân nhỏ thành các quyền:
  - + CREATE: Tạo mới, gồm:
    - CREATE SCHEMA: tạo một lược đồ CSDL (database schema)
    - CREATE TABLE: tạo một bảng dữ liệu hay quan hệ (table hay relation)
    - CREATE VIEW: tạo một khung nhìn (view)
  - + ALTER: chỉnh sửa, cập nhật một lược đồ hoặc một bảng
  - + DROP: xóa một bảng hoặc một khung nhìn
  - + SELECT: quyền truy vấn, lấy thông tin trong CSDL
  - + MODIFY: quyền thêm/xóa/sửa dữ liệu (record/tuple)
  - + REFERENCE: xác định các khóa ngoại và các cột tham chiếu trong các bảng khác.

## 2.2. Điều khiển truy cập tùy ý

### ▪ Quy tắc trao quyền trong DAC (tiếp):

- *Quyền ở cấp đối tượng CSDL (object level)*: là những quyền trên mỗi đối tượng trong hệ CSDL, người dùng tạo ra đối tượng nào thì sẽ có tất cả các quyền trên đối tượng đó.

Ở cấp đối tượng CSDL, các quyền truy cập ở cấp người dùng (CREATE, ALTER, DROP, ...) được áp dụng cho:

- + Bảng cơ sở (table): Tất cả dữ liệu trong bảng (quan hệ)
- + Khung nhìn (view): Tất cả dữ liệu được xác định trong view (bảng ảo)
- + Cột (column): Dữ liệu trong một cột cụ thể

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Một số mô hình DAC:*

- Mô hình ma trận truy cập ACM
- Mô hình Take-Grant
- Mô hình Action-Entity
- Mô hình của Wood và các cộng sự
- ...

## 2.2. Điều khiển truy cập tùy ý

- **Mô hình ma trận truy cập - ACM (Access Control Matrix Model):**
  - Là mô hình bảo mật được dùng cho cả cấp hệ điều hành và cấp CSDL, được đề xuất bởi Lampson (1971), được mở rộng bởi Graham và Denning (1972), sau đó được trình bày lại một cách có hệ thống bởi Harrison và các cộng sự (1976).
  - Ma trận truy cập là ma trận giữa các chủ thể (subject), các đối tượng (object) và các quyền tương ứng của chủ thể với đối tượng.

## 2.2. Điều khiển truy cập tùy ý

### ■ *Mô hình ma trận truy cập - ACM (tiếp):*

- *Trạng thái định quyền (Authorization state)*

$$Q = (S, O, A)$$

- + S - Subjects: là tập các chủ thể-các thực thể chủ động (active entity) sử dụng các nguồn tài nguyên của hệ thống (nhóm các người dùng (group), tiến trình (process), ...)
- + O - Objects: là tập các đối tượng-các thực thể cần được bảo vệ (ở mức hệ điều hành: file, bộ nhớ, segments, tiến trình; ở mức CSDL: CSDL, quan hệ, dòng, cột)
- + A (Access matrix): là ma trận truy cập, dòng: các chủ thể  
cột: các đối tượng; mỗi ô  $A[s,o]$  chứa các chế độ truy cập mà chủ thể  $s$  được quyền thao tác trên đối tượng  $o$

## 2.2. Điều khiển truy cập tùy ý

- *Mô hình ma trận truy cập - ACM (tiếp):*
  - *Ma trận truy cập:*

	$O_1$	...	$O_i$	...	$O_m$
$S_1$	$A[s_1, o_1]$		$A[s_1, o_i]$		$A[s_1, o_m]$
...					
$S_i$	$A[s_i, o_1]$		$A[s_i, o_i]$		$A[s_i, o_m]$
...					
$S_n$	$A[s_n, o_1]$		$A[s_n, o_i]$		$A[s_n, o_m]$

## 2.2. Điều khiển truy cập tùy ý

### ■ *Mô hình ma trận truy cập - ACM (tiếp):*

- Ví dụ: Hệ thống có 2 người dùng là *Alice* và *Bob* có quyền truy xuất 3 file: *bill.doc*, *edit.exe* và *fun.com* như sau:
  - + *Alice* không có quyền truy xuất file *bill.doc*, *Bob* có quyền đọc hoặc ghi file này.
  - + *Alice* và *Bob* chỉ có quyền thực thi file *edit.exe*.
  - + *Alice* và *Bob* có quyền thực thi và quyền đọc file *fun.com*, *Bob* có thêm quyền ghi lên file này

Users\Files	bill.doc	edit.exe	fun.com
Alice	{ }	{ execute }	{ execute, read }
Bob	{ read, write }	{ execute }	{ execute, read, write }

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Mô hình ma trận truy cập - ACM (tiếp):*

#### - *Giải pháp cài đặt ma trận ACM:*

- + Cài đặt trực tiếp một ma trận ACM là không khả thi vì nó vừa quá lớn, gây tốn bộ nhớ, mặt khác bộ nhớ đã được cấp phát nhưng không sử dụng => gây lãng phí.
- + Các giải pháp cài đặt ACM khả thi:
  - Danh sách điều khiển truy cập ACL
  - Danh sách khả năng CL
  - Bộ ba điều khiển truy cập ACT
  - Cách tiếp cận khác: sử dụng các khái niệm riêng như Lock và key.



## 2.2. Điều khiển truy cập tùy ý

- *Mô hình ma trận truy cập - ACM (tiếp):*

- *Giải pháp cài đặt ma trận ACM (tiếp):*

- + *Danh sách điều khiển truy cập ACL - Access Control Lists:*  
phân rã ma trận theo cột, gồm một đối tượng và nhiều chủ thể, xác định những chủ thể nào có thể truy cập đối tượng.

*Ví dụ:*

- *File bill.doc  $\rightarrow \{(Bob, \{read, write\})\}$*
- *File edit.exe  $\rightarrow \{(Alice, \{execute\}), (Bob, \{execute\})\}$*
- *File fun.com  $\rightarrow \{(Alice, \{execute, read\}), (Bob, \{execute, read, write\})\}$*

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Mô hình ma trận truy cập - ACM (tiếp):*

#### - *Giải pháp cài đặt ma trận ACM (tiếp):*

+ *Danh sách khả năng CL - Capability Lists*: phân rã ma trận theo dòng, gồm một chủ thể và nhiều đối tượng, xác định chủ thể có thể truy cập đến những đối tượng nào.

*Ví dụ:*

- *Alice*  $\rightarrow \{(file\ edit.exe, \{execute\}), (file\ fun.com, \{execute, read\})\}$
- *Bob*  $\rightarrow \{(file\ bill.doc, \{read, write\}), (file\ edit.exe, \{execute\}), (file\ fun.com, \{execute, read, write\})\}$

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Mô hình ma trận truy cập - ACM (tiếp):*

- *Giải pháp cài đặt ma trận ACM (tiếp):*

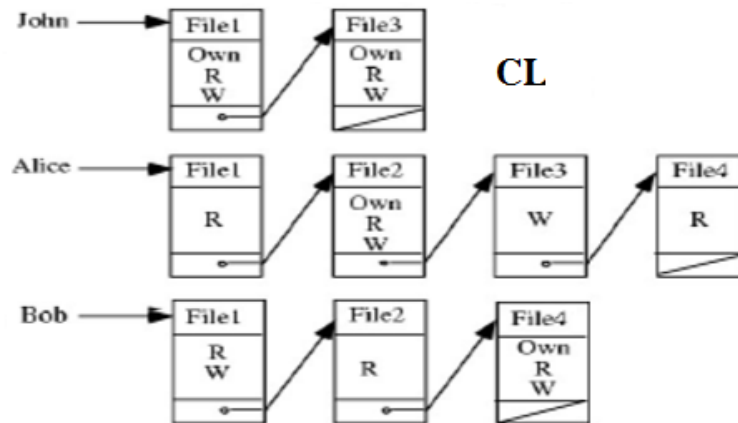
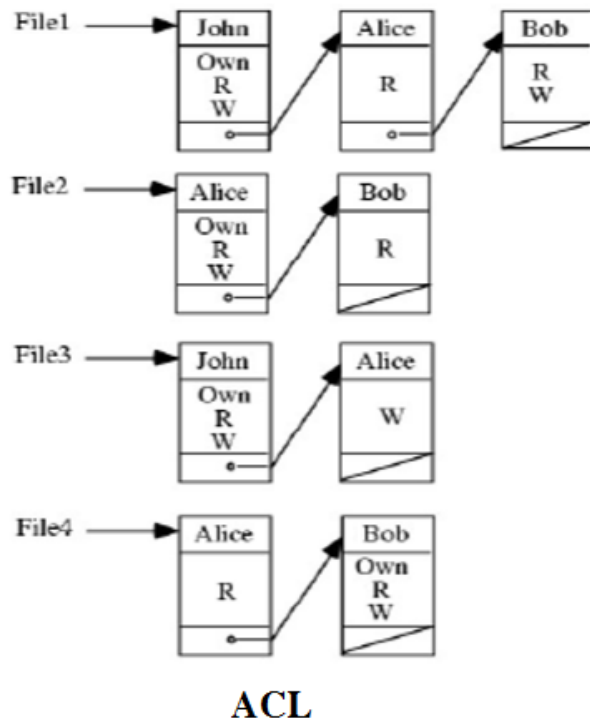
+ *Bộ ba điều khiển truy cập ACT - Access Control Triples: danh sách bộ ba (chủ thể, quyền truy nhập, đối tượng) được lưu trong một cấu trúc bảng, chính là biểu diễn rút gọn của ma trận bằng cách triệt tiêu toàn bộ các ô dữ liệu trống.*

+ *Ví dụ:*

Subject	Access	Object
Alice	execute	edit.exe
Alice	execute	fun.com
Alice	read	fun.com
Bob	read	bill.doc
Bob	write	bill.doc
Bob	execute	edit.exe
Bob	execute	fun.com
Bob	read	fun.com
Bob	write	fun.com

## 2.2. Điều khiển truy cập tùy ý

### ■ Mô hình ma trận truy cập - ACM (tiếp):



ACT

Subject	Access mode	Object
John	Own	File 1
John	R	File 1
John	W	File 1
John	Own	File 3
John	R	File 3
John	W	File 3
Alice	R	File 1
Alice	Own	File 2
Alice	R	File 2
Alice	W	File 2
Alice	W	File 3
Alice	R	File 4
Bob	R	File 1
Bob	W	File 1
Bob	R	File 2
Bob	Own	File 4
Bob	R	File 4
Bob	W	File 4

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Mô hình ma trận truy cập - ACM (tiếp):*

- *Bài tập:* Cho bảng sau, xây dựng ma trận truy cập, danh sách điều khiển truy cập, danh sách khả năng

USER	ACCESS MODE	OBJECT
Ann	own	File 1
Ann	read	File 1
Ann	write	File 1
Ann	read	File 2
Ann	write	File 2
Ann	execute	Program 1
Bob	read	File 1
Bob	read	File 3
Bob	write	File 3
Carl	read	File 2
Carl	execute	Program 1
Carl	read	Program 1

## 2.2. Điều khiển truy cập tùy ý

### ■ *Mô hình Take-Grant:*

- Được đề xuất bởi Johns và các cộng sự (1976)
- Sử dụng các cấu trúc hình học để biểu diễn mối quan hệ về quyền giữa các chủ thể với đối tượng, giữa chủ thể với chủ thể và giữa đối tượng với đối tượng
- Có thể được xem là một dạng mở rộng của mô hình ma trận truy cập

## 2.2. Điều khiển truy cập tùy ý

### ■ *Mô hình Take-Grant (tiếp):*

- *Trạng thái định quyền:*

$$G = (S, O, E)$$

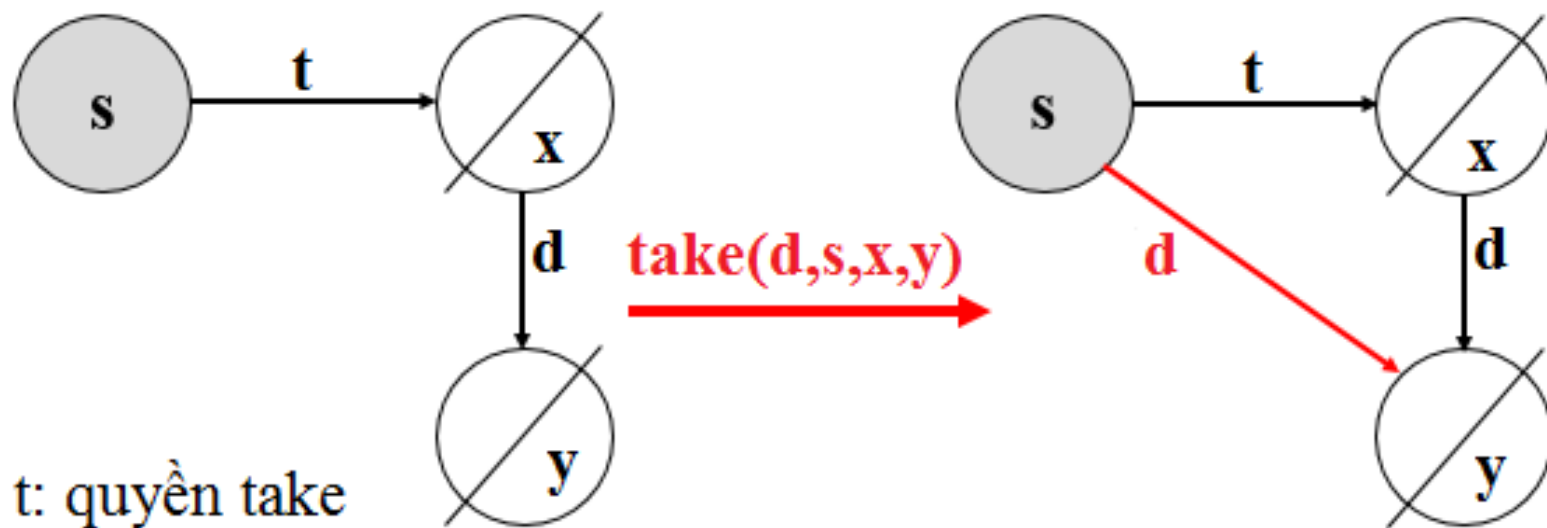
- + S: tập các chủ thể (người dùng, quá trình, chương trình)
- + O: tập các đối tượng bị động (file, bộ nhớ, CSDL, bảng, hàng, trường dữ liệu)
- +  $V = S \cup O$ : tập các đỉnh,  $S \cap O = \emptyset$
- + E: tập các cung được gán nhãn

## 2.2. Điều khiển truy cập tùy ý

### ▪ Mô hình Take-Grant (tiếp):

- Thao tác Take và Grant:

+ **take(d,s,x,y)**: chủ thể  $s$  lấy quyền  $d$  trên đối tượng/chủ thể  $y$  từ đối tượng/chủ thể  $x$



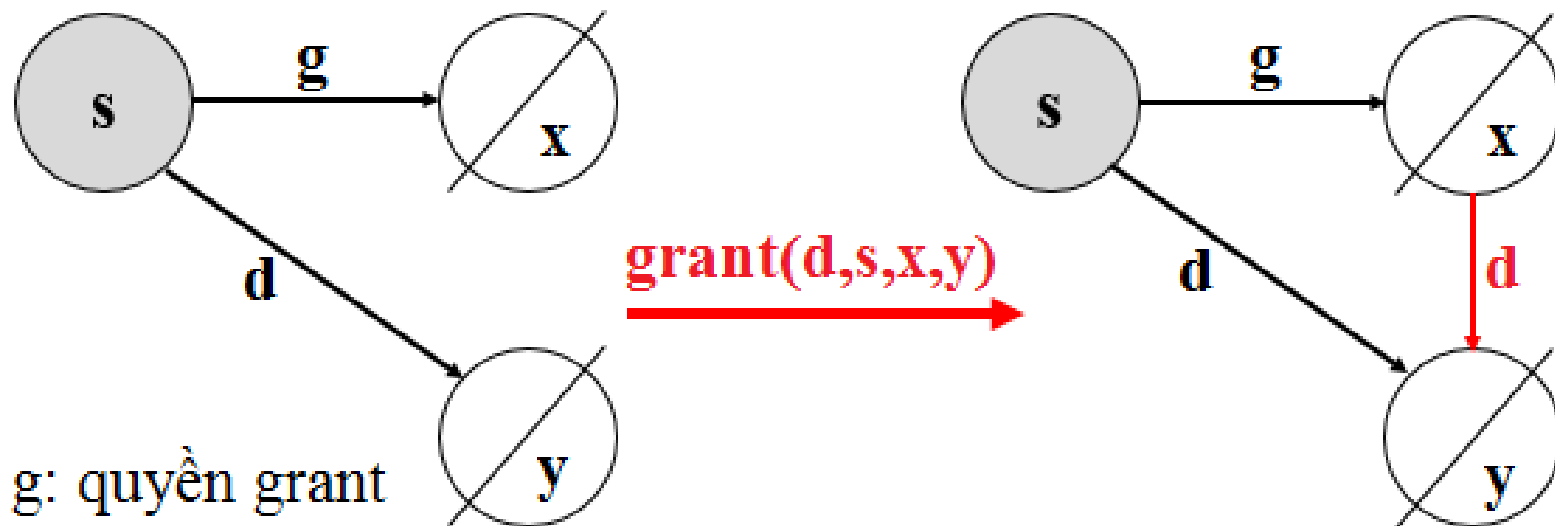


## 2.2. Điều khiển truy cập tùy ý

### ▪ Mô hình Take-Grant (tiếp):

- Thao tác Take và Grant (tiếp):

+ **grant(d,s x y)**: chủ thể  $s$  gán (cấp) quyền  $d$  trên đối tượng/chủ thể  $y$  cho đối tượng/chủ thể  $x$



## 2.2. Điều khiển truy cập tùy ý

### ■ *Mô hình Take-Grant (tiếp):*

#### - *Các loại quyền truy cập:*

+ read, write: không làm thay đổi trạng thái định quyền

+ take, grant: làm thay đổi trạng thái định quyền

#### - *Các loại thao tác truyền quyền:*

+ take, grant: lấy và gán quyền

+ create(s,x): chủ thể  $s$  tạo đối tượng/chủ thể  $x$ , khi đó cung nối giữa  $s$  và  $x$  sẽ được gán nhãn  $p$  (possess: sở hữu)

+ remove<sub>p</sub>(s,x): chủ thể  $s$  bị thu hồi quyền  $p$  trên đối tượng/chủ thể  $x$

## 2.2. Điều khiển truy cập tùy ý

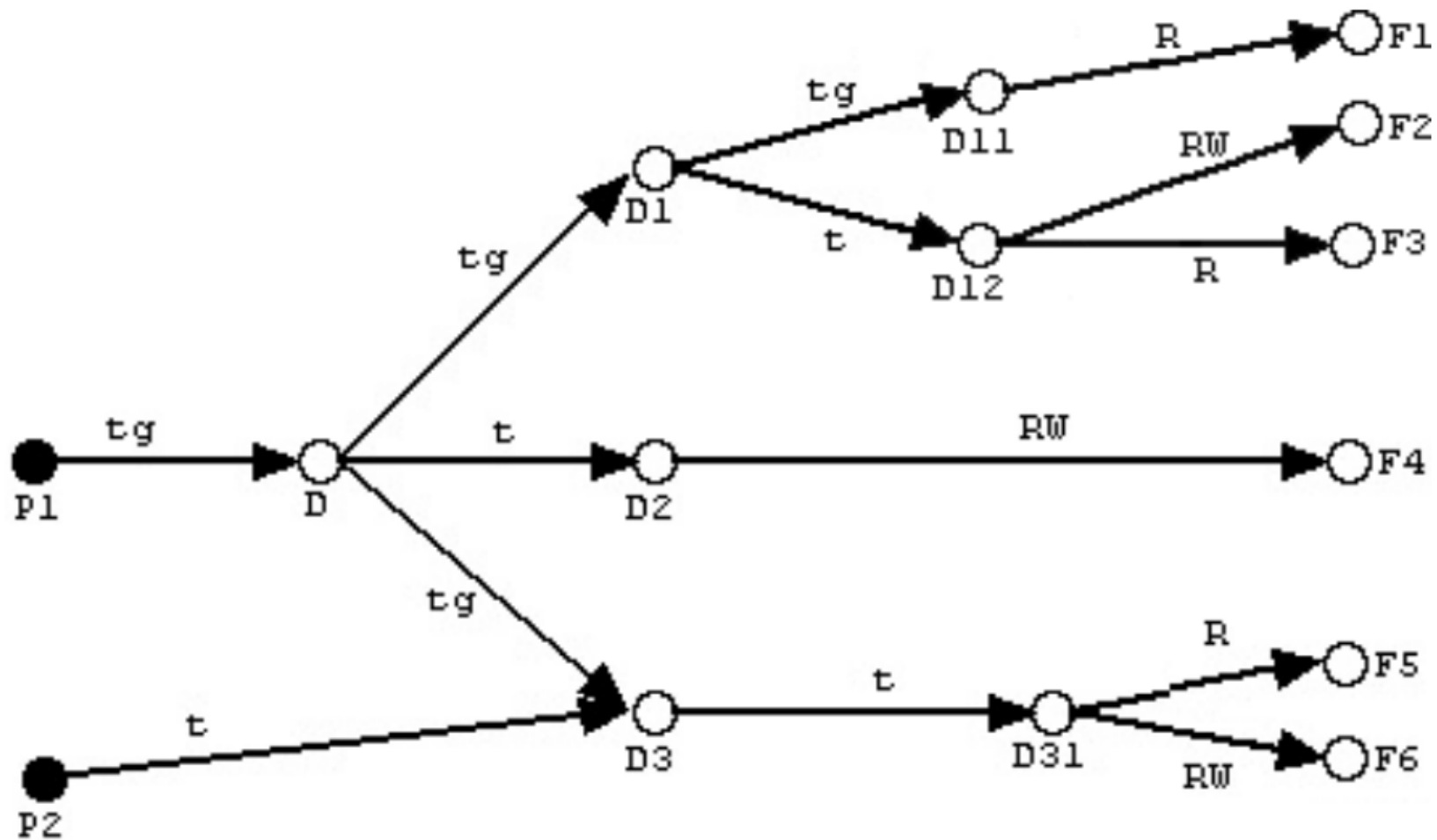
### ■ *Mô hình Take-Grant (tiếp):*

- *Khuyết điểm của mô hình Take-Grant:*
  - + Không có tính chọn lọc của các quyền quản lý:
    - Tất cả các quyền của  $s$  đều có thể bị truyền đi nếu  $s$  sở hữu quyền *GRANT*
    - Tất cả các quyền của  $o/s$  đều có thể bị lấy đi (truyền đi) nếu có một quyền *TAKE* trên nó.
  - + Không quản lý được sự lan truyền quyền
  - + Tính không cục bộ: nếu  $s$  có quyền *GRANT* trên  $o$  thì  $s$  có thể truyền bất kỳ quyền gì của mình cho  $o$ . Như vậy không kiểm soát được tập quyền có thể có trên  $o$ .
  - + Khả năng lan truyền ngược của dòng di chuyển quyền

## 2.2. Điều khiển truy cập tùy ý

### ▪ Mô hình Take-Grant (tiếp):

- Biểu diễn Take-Grant của cấu trúc thư mục:



## 2.2. Điều khiển truy cập tùy ý

- ***Điều khiển dữ liệu với SQL:***
  - ***2 cú pháp lệnh cơ bản:***
    - + GRANT (cấp/trao quyền)
    - + REVOKE (thu hồi/hủy bỏ quyền)

## 2.2. Điều khiển truy cập tùy ý

### ▪ *Điều khiển dữ liệu với SQL:*

- **GRANT:** cấp (trao) quyền quyền trên các đối tượng CSDL (của mình) cho những người dùng khác, cú pháp:

GRANT <danh sách các quyền>

ON <danh sách đối tượng CSDL>

TO <danh sách người dùng khác>

- **REVOKE:** thu hồi (hủy bỏ) những quyền trên các đối tượng CSDL (của mình) từ những người dùng khác, cú pháp:

REVOKE <danh sách các quyền>

ON <danh sách đối tượng CSDL>

FROM <danh sách người dùng khác>

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Lan truyền quyền với *GRANT OPTION*:

- + Giả sử người dùng *A* là chủ sở hữu của bảng *R*, *A* có thể cấp quyền *D* trên *R* cho người dùng *B*, nếu có bổ sung *WITH GRANT OPTION* vào cuối câu lệnh *GRANT* thì *B* được phép cấp quyền *D* trên *R* cho những người dùng khác. Giả sử *B* lại cấp quyền *D* trên *R* cho người dùng *C* và cũng sử dụng *WITH GRANT OPTION*, khi đó *C* tiếp tục được phép cấp quyền *D* trên *R* cho những người dùng khác → lan truyền quyền *D* trên *R* (chủ sở hữu của *R* (người dùng *A*) có thể không biết đến sự lan truyền quyền *D* trên *R*!)
- + Nếu người dùng *A* thu hồi lại quyền *D* đã cấp cho người dùng *B* thì tất cả những quyền *D* lan truyền bắt đầu từ *B* phải bị hệ thống tự động thu hồi lại.

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Lan truyền quyền với GRANT OPTION (tiếp):

Ví dụ: Giả sử Rogers là chủ sở hữu của bảng *EMPLOYEE*, Rogers và các người dùng khác lần lượt thực hiện các lệnh:

<i>Rogers:</i>	GRANT ALL PRIVILEGES ON EMPLOYEE TO Miller WITH GRANT OPTION
<i>Miller:</i>	GRANT ALL PRIVILEGES ON EMPLOYEE TO Chen WITH GRANT OPTION
<i>Chen:</i>	GRANT ALL PRIVILEGES ON EMPLOYEE TO Williams WITH GRANT OPTION
<i>Rogers:</i>	GRANT SELECT ON EMPLOYEE TO Goldstein WITH GRANT OPTION
<i>Goldstein:</i>	GRANT SELECT ON EMPLOYEE TO Rodriguez WITH GRANT OPTION
<i>Rogers:</i>	REVOKE ALL PRIVILEGES ON EMPLOYEE FROM Miller CASCADE

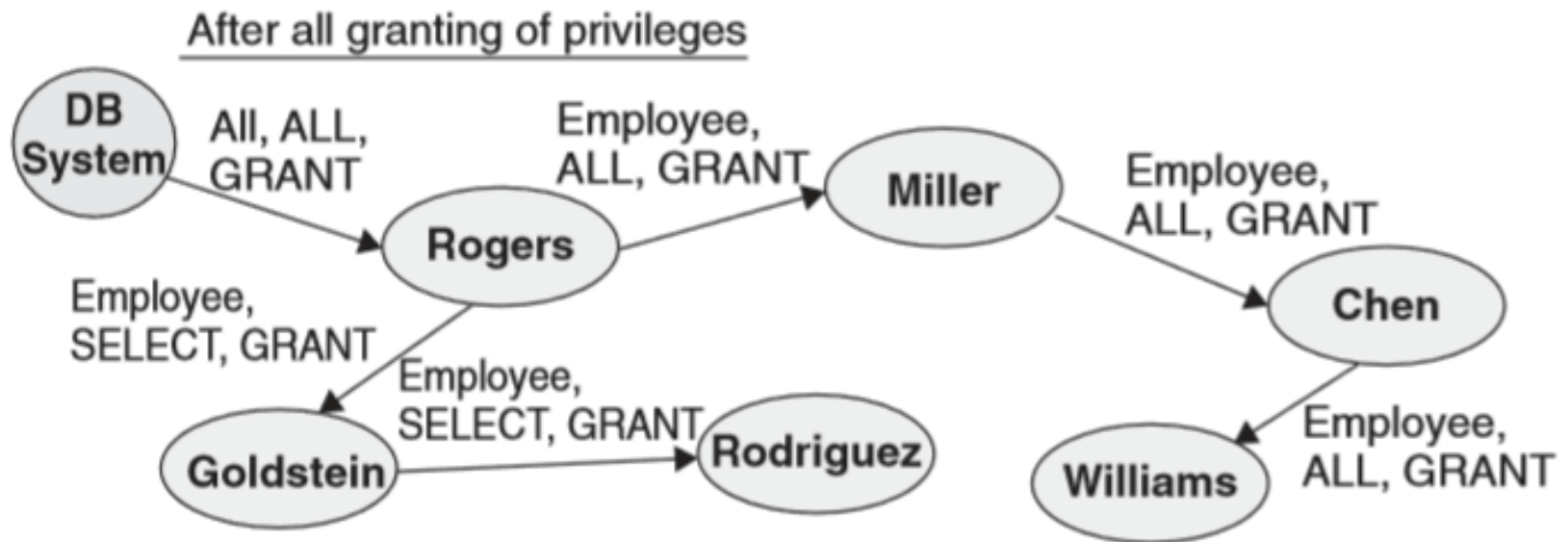


## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Lan truyền quyền với GRANT OPTION (tiếp):

Ví dụ (tiếp): Sau khi cấp quyền:



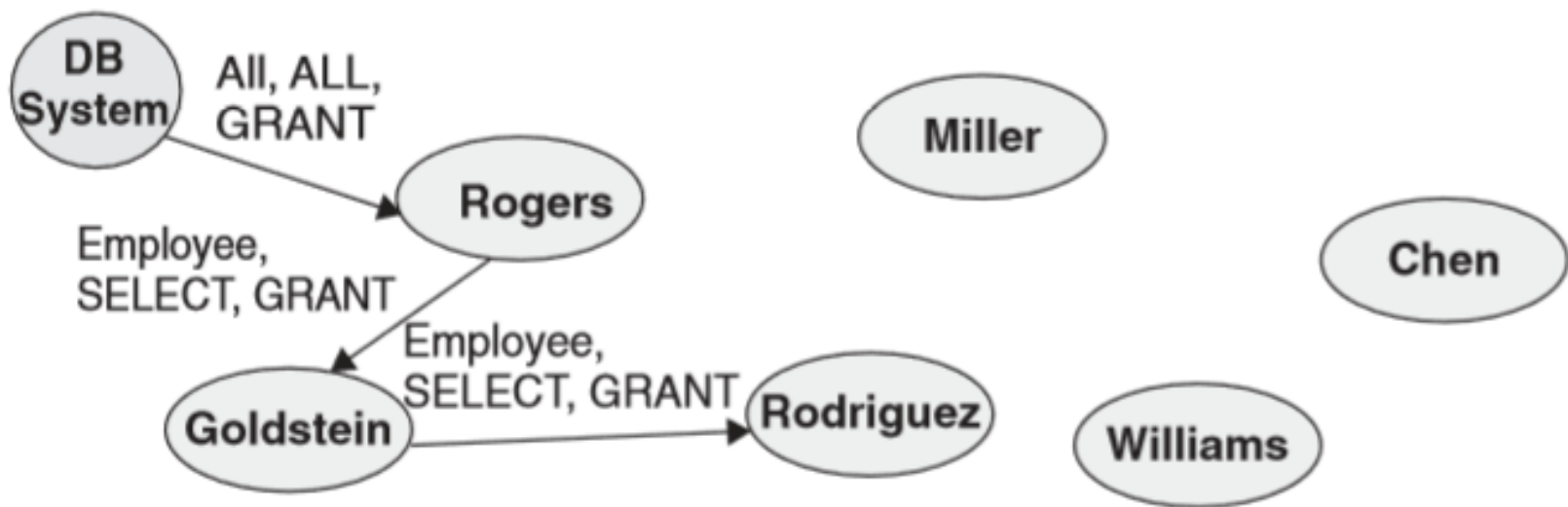
## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Lan truyền quyền với GRANT OPTION (tiếp):

Ví dụ (tiếp): Sau khi thu hồi quyền:

After revoking of privileges from Miller by Rogers with cascade option



## 2.2. Điều khiển truy cập tùy ý

- **Điều khiển dữ liệu với SQL:**

- **Tùy chọn REFERENCES:**

Ví dụ: Giả sử Nash là chủ sở hữu của bảng *DEPARTMENT*

*DEPARTMENT (DeptNo, DeptName, DeptLocation)*

Nash có thể ủy quyền cho Miller tạo một bảng *EMPLOYEE* chứa khóa ngoại liên kết với cột *DeptNo* trong bảng *DEPARTMENT*, việc ủy quyền được thực hiện bằng cách Nash cấp cho Miller quyền *REFERENCES* tới cột *DeptNo*:

*EMPLOYEE (EmployeeNo, FirstName, LastName, Address, Phone, Employee Position, Salary, EmployeeCode, DeptNo)*

*Foreign Key: DeptNo REFERENCES DEPARTMENT*

## 2.2. Điều khiển truy cập tùy ý

- **Điều khiển dữ liệu với SQL:**

- **Tùy chọn REFERENCES (tiếp):**

Ví dụ (tiếp): Nếu *Miller* mất quyền *REFERENCES* tới cột *DeptNo* trong bảng *DEPARTMENT*, ràng buộc khóa ngoại trong bảng *EMPLOYEE* sẽ bị hủy bỏ.

Giả sử *Miller* có quyền *SELECT* trên cột *DeptNo* của bảng *DEPARTMENT*, không phải quyền *REFERENCES*, khi đó, *Miller* sẽ không được phép tạo bảng *EMPLOYEE* với khóa ngoại liên kết tới cột *DeptNo* trong bảng *DEPARTMENT*.

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Tùy chọn REFERENCES (tiếp):

Ví dụ (tiếp): Tại sao không cấp cho *Miller* quyền *SELECT* và cho phép *Miller* tạo bảng *EMPLOYEE* với một cột khóa ngoại liên kết tới cột *DeptNo* trong bảng *DEPARTMENT*?

Nếu như vậy, giả sử *Miller* tạo bảng với ràng buộc khóa ngoại:

*EMPLOYEE* (*EmployeeNo*, *FirstName*, *LastName*, *Address*, *Phone*, *Employee Position*, *Salary*, *EmployeeCode*, *DeptNo*)

*Foreign Key: DeptNo REFERENCES DEPARTMENT ON DELETE NO ACTION*

Với tùy chọn *NO ACTION* trong khai báo khóa ngoại, *Nash* bị cấm xóa các hàng trong bảng *DEPARTMENT* mặc dù *Nash* là chủ sở hữu của bảng này!

## 2.2. Điều khiển truy cập tùy ý

- **Điều khiển dữ liệu với SQL:**
  - **Tùy chọn REFERENCES (tiếp):**
    - khi cần hạn chế cấp quyền → sử dụng REFERENCES (quyền SELECT chỉ nhằm mục đích cho phép đọc các giá trị).

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Sử dụng khung nhìn (Views):

- + Một khung nhìn của người dùng (View) giống như một mô hình cá nhân hóa của CSDL được điều chỉnh cho từng nhóm người dùng. Nếu một nhóm người dùng, giả sử, trong department là marketing, chỉ cần truy cập vào một số cột của bảng *DEPARTMENT* và *EMPLOYEE*, thì có thể tạo View chỉ bao gồm các cột đó.
- + View không giống như table, không lưu trữ dữ liệu thực tế. View là bảng ảo. Khi người dùng truy cập dữ liệu thông qua View, họ sẽ chỉ nhận được dữ liệu từ các mục, các bảng được xác định trong view.
- ➔ View cung cấp một phương pháp đơn giản để cấp quyền truy cập theo cách cá nhân hóa và được xem là một công cụ bảo mật mạnh mẽ.

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Sử dụng khung nhìn (Views):

- + *Chú ý:* Để người dùng tạo view từ nhiều bảng, người dùng phải có quyền truy cập trên các bảng đó. View tự động bị hủy bỏ nếu quyền truy cập bị hủy bỏ.
- + *Ví dụ:* Cấp quyền truy cập cho *Miller* để đọc các thông tin *EmployeeNo*, *FirstName*, *LastName*, *Address*, và *Phone* của các nhân viên trong phòng ban nơi *Miller* làm việc:

```
CREATE VIEW MILLER
```

```
AS
```

```
SELECT EmployeeNo, FirstName, LastName, Address, Phone
```

```
FROM EMPLOYEE
```

```
WHERE DeptNo = (SELECT DEPARTMENT.DeptNo
```

```
WHERE DEPARTMENT.DeptNo = EMPLOYEE.DeptNo
```

```
AND (EMPLOYEE.LastName = 'Miller'));
```

```
GRANT SELECT ON MILLER TO Miller;
```



## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Một số ví dụ:

+ DBA cấp cho Miller quyền tạo lược đồ:

*GRANT CREATETAB TO Miller;*

+ Miller định nghĩa lược đồ, bắt đầu với lệnh tạo lược đồ:

*CREATE SCHEMA EmployeeDB AUTHORIZATION Miller;*

(Sau đó Miller sử dụng các cú pháp định nghĩa dữ liệu (DDL) để tạo các bảng *DEPARTMENT* và *EMPLOYEE*)

+ Miller cấp cho Rodriguez quyền chèn dữ liệu vào cả 2 bảng:

*GRANT INSERT ON DEPARTMENT, EMPLOYEE TO Rodriguez*

## 2.2. Điều khiển truy cập tùy ý

### ▪ Điều khiển dữ liệu với SQL:

#### - Một số ví dụ:

+ Miller cấp cho Goldstein quyền chèn, xóa các hàng trong cả 2 bảng và cho phép Goldstein lan truyền các quyền này:

*GRANT INSERT,DELETE ON DEPARTMENT,EMPLOYEE TO Goldstein WITH GRANT OPTION;*

+ Goldstein cấp quyền chèn và xóa các hàng trong bảng *DEPARTMENT* cho Rogers:

*GRANT INSERT, DELETE ON DEPARTMENT TO Rogers;*

+ Miller cấp cho Williams quyền chỉ cập nhật các cột *Salary*, *EmployeePosition* trong bảng *EMPLOYEE*:

*GRANT UPDATE ON EMPLOYEE (Salary, EmployeePosition) TO Williams*

## 2.2. Điều khiển truy cập tùy ý

- **Điều khiển dữ liệu với SQL:**

- **Một số ví dụ:**

- + DBA cấp cho Shady quyền tạo bảng:

- GRANT CREATETAB TO Shady;*

- + Shady tạo bảng *MYTABLE* và sau đó cấp cho Miller quyền chèn các hàng vào trong bảng *MYTABLE*:

- GRANT INSERT ON MYTABLE TO Miller;*