# CTT201

# An toàn và Bảo mật Dữ liệu trong HTTT

## Chương 2: Điều khiển truy cập (Access Control – AC)

## Phần 3 - MAC

TS. Phạm Thị Bạch Huệ
Ths. Hoàng Anh Tú

Khoa Công nghệ thông tin – Đại học Khoa học tự nhiên

# Có 3 kiểu điều khiển truy cập

- DAC (Discretionary Access Control)
  - Cho biết chủ thể nào có thể truy cập kiểu gì đến các đối tượng CSDL.
  - Có những nguyên tắc để 1 chủ thể có thể tùy ý cấp quyền hay lấy lại quyền cho/ từ 1 chủ thể khác.
- 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.
- RBAC (Role-based Access Control)
  - Vai trò là 1 tập các quyền. 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 thuộc vai trò đó.

# Nội dung

- MAC
- Mô hình dữ liệu MRL

# MAC

# MAC

- MAC (*Mandatory Access Control*)

- Điều khiển truy xuất dựa trên sự phân lớp chủ thể truy xuất (*Subject*) và đối tượng dữ liệu (*Object*).
- MAC được dùng trong môi trường cần tính chất an ninh cao như: chính phủ, quân đội, …

- MAC được ORACLE cài đặt.

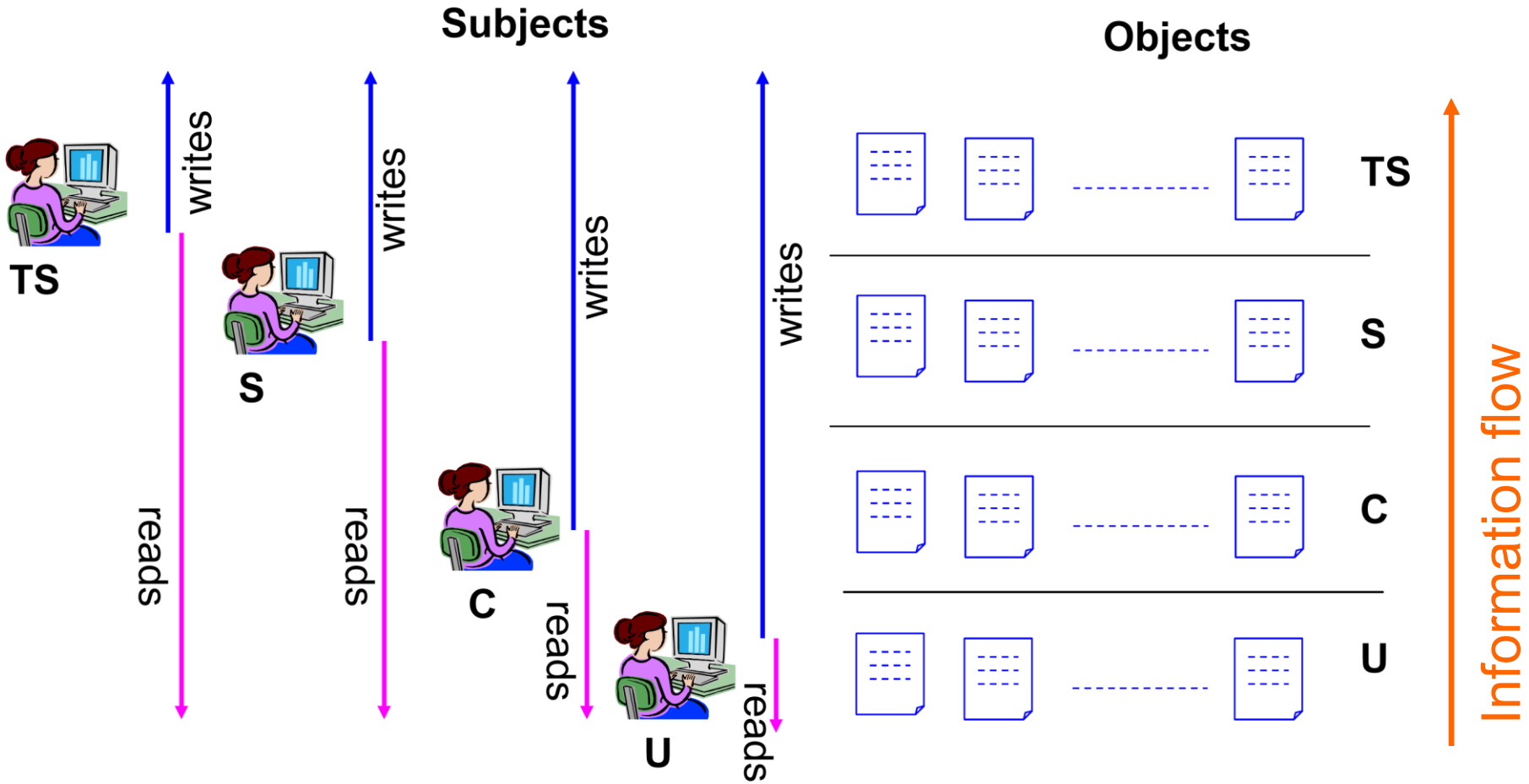
# MAC

- **Đối tượng dữ liệu (Object):** tables, views, tuples.
- **Chủ thể truy cập (Subject):** users, user programs.

- **Security class (or level, or labels)**
  - ✓ Top Secret (TS), Secret (S), Confidential (C), Unclassified (U)
  - ✓ Trong đó: TS > S > C > U

- Mỗi chủ thể và mỗi đối tượng được xếp vào một trong các Security Class:
  - ✓ No read – up: Chủ thể S có thể Đọc đối tượng O nếu Class(S) >= Class(O).
  - ✓ No write – down: Chủ thể S có thể Ghi đối tượng O nếu Class(S) <= Class(O).
- Tuy nhiên, thực tế các hệ thống không cho phép write up, mà chỉ cho phép write cùng mức. Hãy kiểm tra điều này trên Oracle?

# MAC



# MAC – Nhận xét

- Nguyên lý về đơn vị dữ liệu của đối tượng bảo mật.
  - Là toàn bộ CSDL, hay tập tin, ở các thuộc tính hay từng item.
- Không có kỹ thuật tự động cho việc gán nhãn bảo mật.
- Nhiều người cùng truy cập tại một thời điểm.
  - Vì áp dụng chính sách dòng thông tin nên những người có mức bảo mật cao hơn bị hạn chế ghi xuống các hạng mục dữ liệu có sự phân loại bảo mật thấp hơn.

- **Ví dụ:**
  - Chủ thể s1 và s2 có nhãn(s1)>nhãn(s2)
  - Mục dữ liệu d với nhãn(d) = nhãn(s2),
  - Luật thương mại cho rằng để ghi dữ liệu lên d của s2 cần sự chấp thuận của s1. Điều này thì không thích hợp cho các ứng dụng thương mại của công nghệ CSDL  MLS.

# MLR

# MLS - MLR

- MAC còn được gọi là điều khiển truy cập đa cấp (Multi-Level Security – MLS), ứng dụng trên CSDL quan hệ, có CSDL quan hệ đa cấp (Multi-Level Relational Model - MLR).

- Hệ thống quản lý dữ liệu đáp ứng các thuộc tính của việc bảo mật đa cấp được thiết kế dựa trên mô hình nền tảng là Bell và LaPadula.

# MLR

- Trong mô hình dữ liệu đa cấp, những mục dữ liệu và chủ thể có lớp truy cập riêng của chúng (hay các mức), ví dụ TS(Top Secret), S(Secret), U(Unclassified) v.v… gồm sự phân loại và sự cho phép sử dụng thông tin bí mật (clearance).

- Chủ thể khi truy cập bị giới hạn bởi những điều khiển truy cập bắt buộc, là "*no read up, no write down*", theo mô hình của Bell và LaPadula.

# MLR

- Một quan hệ đa cấp được mô tả bởi hai thành phần:
- R($A_1$,$C_1$,…., $A_n$,$C_n$, TC) trong đó:
  - $A_i$ là một thuộc tính trong miền $D_i$.
  - $C_i$ là một thuộc tính phân loại cho $A_i$; miền của nó là một tập hợp của lớp truy cập mà có thể được kết hợp với giá trị của $A_i$.
  - TC là thuộc tính phân loại của bộ. (TC=*TUPLE-CLASS*), là lớp truy cập lớn nhất trong các $C_i$.
- Thuộc tính phân loại không chấp nhận giá trị rỗng.

| Name | $C_{Name}$ | Dept# | $C_{Dept\#}$ | Salary | $C_{Dept\#}$ | TC |
|---|---|---|---|---|---|---|
| **A** | Low | **Dept1** | Low | **100K** | Low | Low |
| **B** | High | **Dept2** | High | **200K** | High | High |
| **S** | Low | **Dept1** | Low | **150K** | High | High |

# MLR

- Thể hiện quan hệ tại lớp c chứa tất cả dữ liệu mà chủ thể tại lớp c thấy được. Do đó, nó chứa tất cả dữ liệu mà các lớp truy cập ≤ c.
- Tất cả các phần tử với lớp truy cập cao hơn c, hoặc không thể so sánh được thì được che giấu bởi giá trị rỗng (*null*)

| Name | $C_{Name}$ | Dept# | $C_{Dept\#}$ | Salary | $C_{Dept\#}$ | TC |
|---|---|---|---|---|---|---|
| **Bob** | Low | **Dept1** | Low | **100K** | Low | Low |
| **Sam** | Low | **Dept1** | Low | **null** | Low | Low |

**Low instance**

| Name | $C_{Name}$ | Dept# | $C_{Dept\#}$ | Salary | $C_{Dept\#}$ | TC |
|---|---|---|---|---|---|---|
| **Bob** | Low | **Dept1** | Low | **100K** | Low | Low |
| **Ann** | High | **Dept2** | High | **200K** | High | High |
| **Sam** | Low | **Dept1** | Low | **150K** | High | High |

**High instance**

# MLR

Các điều kiện bắt buộc:

- **Quan hệ đa cấp phải thỏa mãn các điều kiện sau:**
  - Với mỗi bộ trong một quan hệ đa cấp, các thuộc tính của khóa chính phải có cùng lớp truy cập.
  - Với mỗi bộ trong một quan hệ đa cấp, lớp truy cập kết hợp với một thuộc tính không phải là khóa phải lớn hơn hoặc bằng lớp truy cập của khóa chính.

- **Các khóa và đa thể hiện:**
  - Trong mô hình quan hệ chuẩn, mỗi bộ được xác định duy nhất bởi  khóa của nó.
  - Khi kết hợp với lớp truy cập, có thể có đồng thời các bộ với giá trị như nhau tại các thuộc tính khóa *nhưng* với sự phân loại khác nhau, hiện tượng này được gọi là đa thể hiện.

# MLR – Đa thể hiện

- **Đa thể hiện xảy ra theo hai trạng thái sau:**
  - *Đa thể hiện vô hình*: Khi một người sử dụng ở mức thấp chèn dữ liệu vào một trường (field) mà đã chứa dữ liệu tại mức cao hơn hay mức không thể so sánh được.
  - *Đa thể hiện hữu hình*: Khi một người sử dụng ở mức cao chèn dữ liệu vào một trường (field) mà đã chứa dữ liệu tại mức thấp hơn.

| Name | CName | Dept# | CDept# | Salary | CDept# | TC |
|---|---|---|---|---|---|---|
| **A** | Low | Dept1 | Low | 100K | Low | Low |
| **B** | High | Dept2 | High | 200K | High | High |
| **S** | Low | Dept1 | Low | 150K | High | High |
| **B** | Low | Dept1 | Low | 100K | Low | Low |

Các bộ khóa là "B" là đa thể hiện

# MLR – Đa thể hiện vô hình

- Giả sử một người sử dụng ở mức thấp yêu cầu chèn một bộ với khóa chính giống nhau tại một bộ tồn tại ở mức cao hơn; DBMS có ba lựa chọn:
  1. Thông báo cho người dùng rằng một bộ với khóa chính giống nhau đã tồn tại ở mức bảo mật cao và từ chối chèn vào.
  2. Thay thế bộ tồn tại ở mức cao hơn với bộ mới được chèn ở mức thấp hơn.
  3. Chèn bộ mới ở mức thấp hơn mà không thay đổi bộ tồn tại ở mức cao hơn (tức là thực thể đa thể hiện).

- Chọn 2) cho phép người sử dụng ở mức thấp ghi đè dữ liệu mà anh ta không nhìn thấy và vì vậy làm mất đi tính toàn vẹn.
- Chọn 3) là một lựa chọn hợp lý; vì tầm quan trọng của nó là giới thiệu một thực thể đa thể hiện.

# MLR – Đa thể hiện hữu hình

- Giả sử một người sử dụng ở mức cao yêu cầu chèn một bộ với khóa chính giống nhau tại một bộ tồn tại ở mức thấp hơn; DBMS có ba lựa chọn:
  1. Thông báo cho người sử dụng rằng một bộ với khóa chính tồn tại giống nhau và từ chối chèn vào.
  2. Thay thế bộ tồn tại ở mức thấp hơn với bộ mới được chèn ở mức cao hơn.
  3. Chèn bộ mới ở mức cao hơn mà không thay đổi bộ tồn tại ở mức thấp hơn (tức là thực thể đa thể hiện).

- Chọn 3) là một lựa chọn hợp lý; vì tầm quan trọng của nó là giới thiệu một thực thể đa thể hiện.

# MLR

- Gồm 5 ràng buộc:
  - Entity integrity (tính toàn vẹn thực thể)
  - Polyintantiation integrity (tính toàn vẹn đa thể hiện),
  - Data-borrow integrity (tính toàn vẹn dữ liệu-mượn),
  - Foreign key integrity (tính toàn vẹn khóa ngoại)
  - Referential integrity (tính toàn vẹn quan hệ)

- Và 5 câu lệnh (insert, delete, select, update, UPLEVEL) thao tác trên quan hệ đa cấp.

# Tham khảo

*Ravi Sandhu, Fang Chen, The multilevel Relational (MLR) data Model, ACM, 1998.*

# Câu hỏi

TS. Phạm Thị Bạch Huệ - ptbhue@fit.hcmus.edu.vn

Ths. Hoàng Anh Tú – hatu@fit.hcmus.edu.vn