MỤC LỤC

Chương 1: Tổng quan về an toàn thông tin trong CSDL

Câu 1: Các mối đe dọa có thể đến với CSDL là gì?

- Khai thác dữ liệu trái phép thông qua suy diễn thông tin được phép.
- Sửa đổi dữ liệu trái phép
- Tấn công từ chối dịch vụ
- Ngoài ra các hiểm họa có thể đến từ thảm họa thiên nhiên, lỗi phần cứng, các sai phạm vô ý của con người gây nên.
- Người dùng lạm dụng quyền
- Tấn công leo thang đặc quyền
- …

Câu 2: Tìm hiểu các cấu hình xử lý CSDL (CSDL tập trung, phân tán, Client/Server). Các cấu hình này được áp dụng như thế nào trong thực tế. (Chú ý: nêu rõ đặc điểm – bản chất và vẽ hình minh họa).

*** Mọi ứng dụng cơ sở dữ liệu đều bao gồm 3 thành phần:**

- Thành phần xử lý ứng dụng
- Thành phần phần mềm cơ sở dữ liệu (DBMS)
- Bản thân cơ sở dữ liệu (DB)

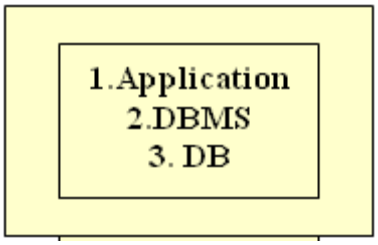
*** Các mô hình xử lý cơ sở dữ liệu phụ thuộc vào định vị của 3 thành phần trên, có 3 mô hình chính là:**

- Mô hình cơ sở dữ liệu tập trung (Centralized database model)

- Mô hình cơ sở dữ liệu phân tán (Distributed database model)
- Mô hình cơ sở dữ liệu Client/Server (Client/Server database model)

*** Mô hình cơ sở dữ liệu tập trung:**

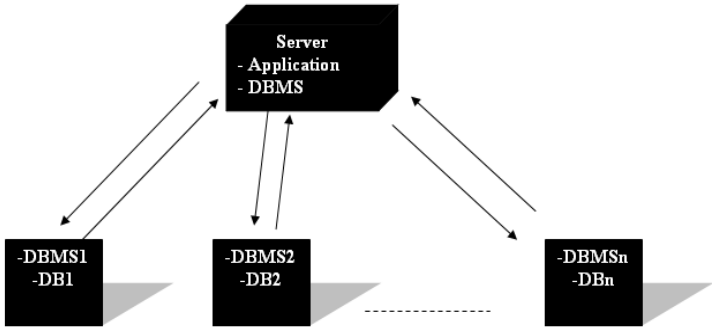
- Cả 3 thành phần: xử lý ứng dụng, phần mềm cơ sở dữ liệu và bản thân cơ sở dữ liệu đều nằm trên một máy.
- Ví dụ các ứng dụng trên một máy sử dụng phần mềm cơ sở dữ liệu Oracle, cơ sở dữ liệu nằm trong máy đó.



*** Mô hình cơ sở dữ liệu phân tán:**

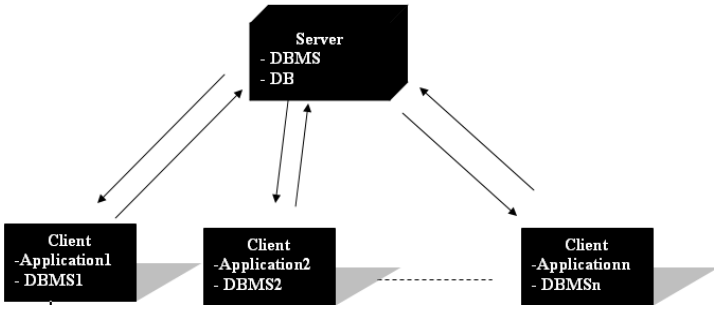
- Trong mô hình này, cơ sở dữ liệu nằm trên nhiều máy khác nhau. Khi máy chủ cần truy xuất dữ liệu, nó sẽ gọi đến các máy này, và cần quá trình đồng bộ dữ liệu

-Mô hình này phù hợp cho các công ty có nhiều chi nhánh khác



*** Mô hình cơ sở dữ liệu Client/Server**

- Trong mô hình này cơ sở dữ liệu nằm trên một máy gọi là Server. Các thành phần xử lý ứng dụng nằm trên các máy Client.



Câu 3: Khái niệm kiểm soát luồng. Cho ví dụ về kiểm soát luồng.

-***Một luồng*** giữa đối tượng X và đối tượng Y xuất hiện khi có một lệnh đọc (*read*) giá trị từ X và ghi (*write*) giá trị vào Y

-***Kiểm soát luồng*** *là kiểm tra xem thông tin trong một số đối tượng có đi vào các đối tượng có mức bảo vệ thấp hơn hay không* **.**Nếu điều này xảy ra thì rõ ràng thông tin ở đối tượng có mức nhạy cảm cao đã bị tiết lộ xuống đối tượng có mức thấp hơn.
-Kiểm soát luồng thông tin trong CSDL thường áp dụng với các CSDL nhiều mức. Việc sao chép dữ liệu từ X tới Y là một ví dụ điển hình về luồng thông tin (từ X tới Y). Ví dụ là một bài test phép toán trong X: bằng cách quan sát kết quả của bài test này có thể suy diễn ra các giá trị của X.

Câu 4: Thế nào là chính sách đặc quyền tối thiểu, chính sách đặc quyền tối đa, và ưu nhược điểm của chúng. Áp dụng chính sách đặc quyền tối thiểu trong kiểm soát luồng thông tin.

Chính sách đặc quyền tối thiểu**:** còn được gọi là chính sách **(*need-to-know*)**. Theo chính sách này, các chủ thể của hệ thống chỉ được sử dụng một lượng thông tin tối thiểu cần cho hoạt động của họ.
-Mỗi đối tượng an toàn -object sẽ được gắn một compartment (chứa nội dung của nó).
-Mỗi chủ thể -subject được phép truy nhập vào một đối tượng nếu *nhu cầu tối thiểu (NTK)* của anh ta phải vượt quá nội dung của đối tượng đó
Nhược điểm:

- Việc ước tính lượng thông tin tối thiểu này là rất khó.
- Những hạn chế truy nhập thông tin có thể vô ích đối với các chủ thể vô hại.

Chính sách đặc quyền tối đa**:**
-Dựa vào nguyên tắc *"khả năng sẵn sàng tối đa"* của dữ liệu, để có thể chia sẻ dữ liệu đến mức tối đa.
-Chính sách này phù hợp với các môi trường như: trường đại học, trung tâm nghiên cứu, là những nơi cần trao đổi dữ liệu, không cần bảo vệ nghiêm ngặt.

Câu 5: Nêu rõ đặc điểm của kiểm soát truy nhập MAC và DAC trong CSDL, nêu sự khác nhau giữa chúng.

Đặc điểm của kiểm soát truy nhập MAC:
Được áp dụng cho các thông tin có yêu cầu bảo vệ nghiêm ngặt, hạn chế truy nhập của các chủ thể vào các đối tượng bằng cách sử dụng các ***nhãn an toàn (label).***
Ví dụ:

Đặc điểm của kiểm soát truy nhập DAC:

+ Chỉ rõ những đặc quyền mà mỗi chủ thể có thể có được trên các đối tượng và trên hệ thống (object prilvilege, system prilvilege).

+ Các yêu cầu truy nhập được kiểm tra, thông qua một cơ chế kiểm soát tuỳ ý, truy nhập chỉ được trao cho các chủ thể thoả mãn các quy tắc cấp quyền của hệ thống.

+ *Được định nghĩa trên một tập*

- Các đối tượng an toàn (security objects)
- Các chủ thể an toàn (security subjects)
- Và các đặc quyền truy nhập (access prilvilege)
  (Quyền truy nhập gồm: object prilvilege, system prilvilege).

+ Người dùng có thể bảo vệ dữ liệu mà họ sở hữu

+ Người chủ sở hữu (owner) có thể gán quyền truy nhập (read, write, execute…) tới các user khác.

+ Việc gán và thu hồi quyền truy nhập là "tùy ý" do những người dùng này.

*****Sự khác nhau giữa MAC và DAC***

MAC	**DAC**
- Kiểm soát quyền dựa vào các nhãn an toàn gắn với chủ thể và đối tượng	- Kiểm soát quyền dựa trên quyền sở hữu đối tượng
- Việc trao, hủy bỏ quyền chỉ do một nhân viên an toàn	- Việc trao, hủy bỏ quyền là túy ý với những user có đặc quyền
- User không thể thay đổi nhãn hay quyền, chỉ do một nhân viên an toàn cao nhất.	- User có thể thay đổi quyền tùy vào đặc quyền của user đó.
- Dùng cho các hệ thống yêu cầu bảo vệ nghiêm ngặt như: quân sự, quốc phòng	- Dùng được cho mọi hệ thống,
- Độ an toàn cao nhưng phức tạp	- Linh hoạt, nhưng độ an toàn không cao

Ví dụ một số hệ quản trị

- Có chính sách DAC như: Access, MySQL, SQL Server, Oracle
- Có chính sách MAC như: Oracle, DB2, Sybase

Câu 6: Tìm hiểu 2 mô hình an toàn là Bell-Lapadula và mô hình RBAC

*** Mô hình Bell- Lapadula**

- Xuất hiện năm 1975, do quân đội Mỹ
- Phù hợp sử dụng trong các hệ thống của quân đội và chính phủ

- Mục đích: đảm bảo tính bí mật

+ Đây là mô hình chính tắc đầu tiên về điều khiển luồng thông tin
+ Là một mô hình tĩnh: mức an toàn (nhãn an toàn) không thay đổi

Người dùng được phân mức độ an toàn, KH: ***Clear(S)***

Đối tượng được phân mức độ nhạy cảm KH: Class (O)

- Thuộc tính an toàn đơn giản (Not Read up):

Một chủ thể S được phép truy nhập đọc đến một đối tượng O chỉ khi ***Clear (S)>=class(O)***

- Thuộc tính * (Not Write down):

Một chủ thể S được phép truy nhập ghi lên một đối tượng O chỉ khi ***Clear (S) <= class(O).***

Ưu điểm: các nhãn an toàn của các chủ thể và các đối tượng không bao giờ được thay đổi trong suốt thời gian hệ thống hoạt động.

Hạn chế:

– *Mới chỉ quan tâm tới tính bí mật*
– *Chưa chỉ ra cách thay đổi các quyền truy nhập cũng như cách tạo và xóa các chủ thể cũng như các đối tượng*

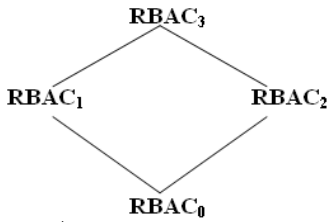
Mô hình này thường áp dụng cho những loại cơ sở dữ liệu quan trọng, cần mức bảo mật cao như trong quân sự hay an ninh quốc phòng…

***Mô hình RBAC(Role based Access Control)**

- Được áp dụng vào đầu những năm 1970s.
- Khái niệm chính của RBAC là những quyền hạn được liên kết với những vai
- trò.
- Khi số lượng chủ thể và đối tượng lớn số lượng quyền hạn có thể trở nên vô
- cùng lớn.
- Nếu người dùng có nhu cầu cao, số lượng cấp và thu hồi quyền diễn ra thường
- xuyên.
- Với RBAC thì có thể giới hạn trước các mối quan hệ vai trò – quyền hạn, làm
- cho việc phân công người dùng đến các vai trò được xác định trước dễ dàng

- hơn.
- Không có RBAC sẽ khó khăn cho việc xác định quyền hạn nào được quy định
- đến người dùng nào.
- Những người dùng được chỉ định những vai trò thích hợp. Điều này làmđơn
- giản cho việc quản lí quyền hạn.
- Trong một tổ chức, những chức năng công việc khác nhau được phân thành
- những vai trò và người dùng được chỉ định vai trò dựa vào trách nhiệm và năng
- lực của họ.

Mô hình RBAC gồm 4 mô hình: RBAC0 , RBAC1 , RBAC2 , RBAC3.



(a) Mối quan hệ giữa các mô hình RBAC

-Mô hình nền tảng *RBAC0* thì ở dưới cùng, nó là yêu cầu tối thiểu cho bất kỳ hệ thống nào có hỗ trợ RBAC.

-Mô hình RBAC1 , RBAC2 được phát triển từ mô hình RBAC0 nhưng có thêm các điểm đặc trưng cho từng mô hình.

– *RBAC1* thêm vào khái niệm của hệ thống phân cấp vai trò (các trạng thái trong đó vai trò có thể thừa kế quyền hạn từ vai trò khác).

– *RBAC2* thêm vào các ràng buộc (áp dụng ràng buộc để có thể thừa nhận cấu hình của các thành phần khác nhau của RBAC). RBAC1 , RBAC2 không liên quan nhau.

- RBAC3 là mô hình tổng hợp của ba mô hình RBAC0 , RBAC1 và RBAC2.

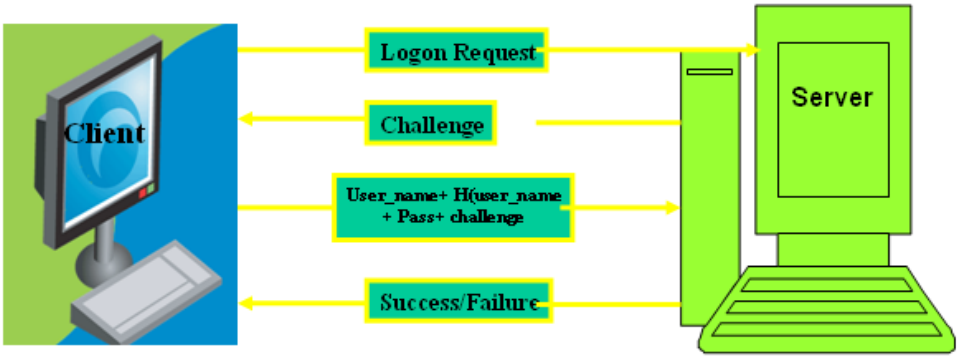
Chương 2 Các cơ chế an toàn cơ bản

Câu 1: Tìm hiểu và mô tả một số phương pháp xác thực hiện nay

****Phương pháp xác thực CHAP**

+ Là mô hình xác thực dựa trên Username/Password.
+ User muốn được xác thực thì nó gửi username đến cho Server.
+ Server sẽ gửi trả lại cho User một thông điệp.
+ Máy tính User sẽ mã hóa thông điệp thử thách đó với khóa là Password và gửi thông điệp đã mã hóa đó trở lại cho Server.

+ Server sẽ so sánh thông điệp mà nó nhận được từ User với thông điệp mà nó mã hóa. Nếu hai thông điệp trùng nhau thì User được xác thực.



Ứng dụng của CHAP hiện nay:

- Thường được sử dụng khi User logon vào các remote servers của công ty.
- Phương pháp này hiện vẫn đang được sử dụng rộng rãi trong nhiều hệ thống server của các công ty lớn

****Phương pháp xác thực Username/ Password**

-Dựa trên sự kết hợp của Username và Password

-Người dùng sẽ được xác thực nếu Username và Password cung cấp có trong CSDL

-Là phương pháp xác thực phổ biến do nó đơn giản và chi phí thấp

-Phương pháp này không bảo mật vì Username và Password truyền dưới dạng rõ trên đường truyền

-Độ phức tạp của MK:

- Các yêu cầu đối với một password "mạnh":
  - Ít nhất 8 ký tự
  - Chứa ít nhất 3 trong số 4 loại ký tự sau:

  + Chữ cái thường
  + Chữ cái hoa
  + Chữ số (0 … 9)
  + Ký tự đặc biệt (!@#$%^&*()_+|~-=\`{}[]:";'<>?,./)
- Không liên quan đến user name, logon name
- Không phải là từ có nghĩa trong từ điển

-PP bảo vệ mật khẩu:

- Password của user được mã hóa để lưu trữ trên server không lưu dưới dạng rõ
- Dùng password một lần (one-time password).
- Kết hợp với trao đổi khóa (giao thức Encrypted Key Exchange_EKE).

**Kerberos

-Dùng một Server trung tâm để kiểm tra việc xác thực user và cấp phát ***vé thông hành (service tickets)*** để User có thể truy cập vào tài nguyên.

-Kerberos là một phương thức rất an toàn trong authentication bởi vì dùng cấp độ mã hóa rất mạnh. Kerberos cũng dựa trên độ chính xác của thời gian xác thực giữa Server và Client Computer.

-Kerberos là nền tảng xác thực chính của nhiều OS như Unix, Windows.

**Tokens

-Tokens là phương tiện vật lý như các thẻ thông minh (smart cards) hoặc thẻ đeo của nhân viên (ID badges) chứa thông tin xác thực.

-Tokens có thể lưu trữ số nhận dạng cá nhân-personal identification numbers (PINs), thông tin về user, hoặc passwords.

-Các thông tin trên token chỉ có thể được đọc và xử lý bởi các thiết bị chuyên dụng, ví dụ như thẻ smart card được đọc bởi đầu đọc smart card gắn trên Computer, sau đó thông tin này được gửi đến Server xác thực.

-Tokens chứa chuỗi text hoặc giá trị số duy nhất thông thương mỗi giá trị này chỉ sử dụng một lần.

** Biometrics (Sinh trắc học):

-Là mô hình xác thực dựa trên đặc điểm sinh học của từng cá nhân, như: Quét dấu vân tay (fingerprint scanner), quét võng mạc mắt (retinal scanner), nhận dạng giọng nói(voice-recognition), nhận dạng khuôn mặt.

-Vì nhận dạng sinh trắc học hiện rất tốn kém chi phí khi triển khai nên không được chấp nhận rộng rãi như các phương thức xác thực khác.

** ***Multi-Factor Authentication*Xác thực đa nhân tố**

-Xác thực nhiều nhân tố dựa trên nhiều nhân tố kết hợp, là mô hình xác thực yêu cầu kiểm tra ít nhất 2 nhân tố xác thực.Có thể đó là sự kết hợp của bất cứ nhân tố nào ví dụ như: bạn là ai, bạn có gì chứng minh, và bạn biết gì?

Ví dụ: về một Multi-Factor Authentication:

Cần phải đưa thẻ nhận dạng vào đầu đọc và cho biết tiếp password là gì

Mutual Authentication:

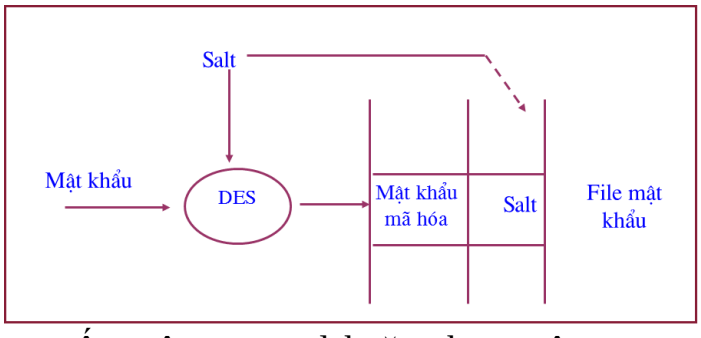
-Xác thực lẫn nhau là kỹ thuật bảo mật mà mỗi thành phần tham gia giao tiếp cần kiểm tra lẫn nhau.

-Trước hết Server chứa tài nguyên kiểm tra "giấy phép truy cập" của client và sau đó client lại kiểm tra "giấy phép cấp tài nguyên" của Server. Điều này giống như khi bạn giao dịch với một Server của ngân hàng, bạn cần kiểm tra Server xem có đúng của ngân hàng đó không hay là một cái bẫy của hacker giăng ra, và ngược lại Server này sẽ kiểm tra lại bạn…

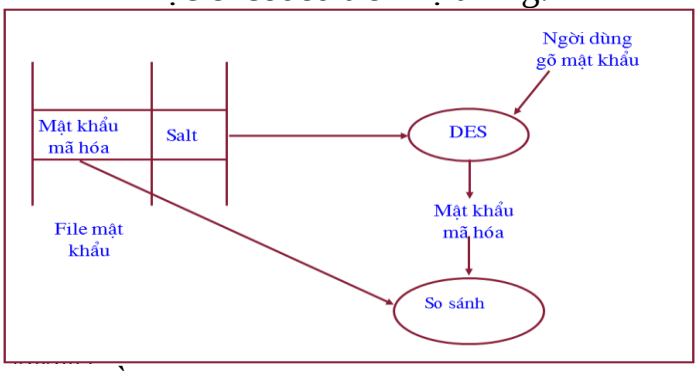
Câu 2: *Salt* là gì? Trong cơ chế bảo vệ mật khẩu, salt làm nhiệm vụ gì?

Một ***salt*** là một số 12 bit đọc thêm vào mật khẩu.

Salt = thoi gian sinh tien trinh + ID tien trinh, ID tien trinh la duy nhat => Salt duy nhất.



Khi user được cung cấp một pasword hoặc chọn một password, password này được encode (DES) với giá trị ngẫu nhiên được gọi là **salt**. Giá trị salt này sẽ được lưu cùng với password đã được encoded trên hệ thống.



Khi người dùng login bằng password của mình, giá trị **salt** sẽ được lấy ra từ encoded password (đã được lưu trên hệ thống) và giá trị salt này sẽ dùng để encode password (mà người dùng vừa gõ vào). Nếu so sánh hai giá trị: đã lưu và vừa được encoded trùng nhau –> người dùng được xác thực.

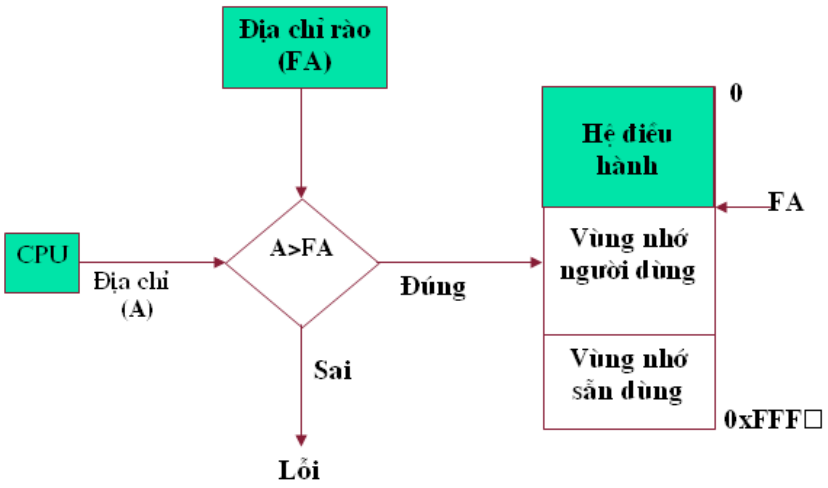
Câu 3: Địa chỉ rào là gì? Ưu nhược điểm.

Địa chỉ rào

+ Địa chỉ rào đánh dấu ranh giới giữa vùng nhớ dành cho hệ điều hành và vùng nhớ cho tiến trình người dùng.
+ Vùng nhớ dành cho hệ điều hành là vùng nhớ thấp
+ Vùng nhớ dành cho người dùng bắt đầu từ địa chỉ rào

Do vùng nhớ cho Hệ điều hành có thể thay đổi, nên dùng một thanh ghi để lưu địa chỉ rào này.

- Cơ chế bảo vệ bộ nhớ dựa vào địa chỉ rào được mô tả trên hình sau:



- ***Ưu điểm:*** bảo vệ được vùng nhớ của hệ điều hành tránh khỏi sự can thiệp của các tiến trình người dùng.

-Nhược điểm:

+ Trong hệ đơn chương: chỉ có một tiến trình người dùng => lãng phí CPU.

+ Trong hệ đa chương: đ/c rào không bảo vệ được vùng nhớ của người dùng này với người dùng khác.

Câu 4: Tái định vị là gì, tái định vị động, tái định vị tĩnh có thể thực hiện trong những thời điểm nào?

Tái định vị:

Tái định vị là quá trình chuyển đổi từ địa chỉ logic trong chương trình sang địa chỉ vật lý.

Địa chỉ vật lý = K + địa chỉ logic

trong đó K là địa chỉ rào của tiến trình người dùng

-Tái định vị có thể thực hiện trong 3 thời điểm :

+ *Thời điểm biên dịch* : Nếu tại thời điểm biên dịch, có thể biết vị trí mà chương trình sẽ thường trú trong bộ nhớ (ví dụ chương trình sẽ có địa chỉ bắt đầu trong bộ nhớ chính là *K1* - địa chỉ rào), thì trình biên dịch có thể phát sinh ngay *mã lệnh thực thi* với các địa chỉ tuyệt đối. Trong suốt quá trình biên dịch, địa chỉ trong chương trình là các *địa chỉ tuyệt đối = địa chỉ tương đối + K1*. Sau đó chương trình nạp, sẽ nạp mã lệnh thực thi này vào vùng nhớ bắt đầu từ K1.

+ *Thời điểm nạp:* Nếu trong quá trình biên dịch cha biết vị trí thường trú của chương trình trong bộ nhớ, thì trình biên dịch sẽ sinh ra mã lệnh thực thi tương đối (object code) chứa các địa chỉ tương đối. Khi nạp chương trình vào bộ nhớ, những địa chỉ tương đối đó sẽ được chuyển thành các địa chỉ tuyệt đối trong bộ

nhớ. Đây đợc gọi là "*tái định vị tĩnh*". Khi có sự thay đổi vị trí lưu trữ tiến trình trong bộ nhớ, chỉ cần nạp lại mà không cần biên dịch lại chương trình.

+ *Thời điểm xử lý:* Nếu có nhu cầu di chuyển tiến trình từ vùng nhớ này sang vùng nhớ khác trong quá trình xử lý – chạy, thì sự kết buộc địa chỉ cần được thực hiện trong thời gian chạy chương trình. Trong trường hợp này, địa chỉ của chương trình khi được nạp vào bộ nhớ cha phải địa chỉ tuyệt đối, nó có thể được tái định vị. Và các địa chỉ đó sẽ được chuyển thành địa chỉ tuyệt đối khi chạy chong trình. Đây được gọi là "*tái định vị động*".

Câu 5: Tìm hiểu 2 cơ chế phân trang, phân đoạn (bảng trang, bảng phân đoạn) và so sánh chúng (chú ý phải vẽ hình minh họa). Thế nào là phân mảnh nội vi, phân mảnh ngoại vi, cho ví dụ?

So sánh 2 cơ chế phân trang và phân đoạn:

***Phân trang:**

- Bộ nhớ vật lý và logic được chia thành các page có kích thước bằng nhau.
- Cơ chế chuyển đổi địa chỉ dùng bảng trang (page table) do hệ điều hành quản lý.
- Cho phép chia sẻ các trang giữa các tiến trình
- Một tiến trình có thể được nạp vào các trang không liên tục nhau.
- Phân mảnh nội vi.

***Phân đoạn:**

- Bộ nhớ vật lý và logic được chia thành các segment có kích thước khác nhau.
- Cơ chế chuyển đổi địa chỉ dùng bảng phân đoạn (segment table) do hệ điều hành quản lý.
- Cho phép chia sẻ các phân đoạn giữa các tiến trình
- Một tiến trình có thể được nạp vào các phân đoạn không liên tục nhau.
- Phân mảnh ngoại vi.

*** Hiện tượng phân mảnh nội vi và ngoại vi**

Hiện tượng phân mảnh nội vi: Khi bộ nhớ được phân phối lớn hơn không đáng kể so với bộ nhớ được yêu cầu của tiến trình, khi đó phần bộ nhớ dư đó sẽ bị lãng phí.

Ví dụ: tiến trình A chỉ yêu cầu 450KB, nhưng lại được cấp 460 KB, do đó là lãng phí mất 10KB.

Hiện tượng phân mảnh ngoại vi là hiện tượng khi các khối nhớ tự do (trong bộ nhớ vật lý) đều quá nhỏ, không đủ để chứa một phân đoạn (trong bộ nhớ logic).

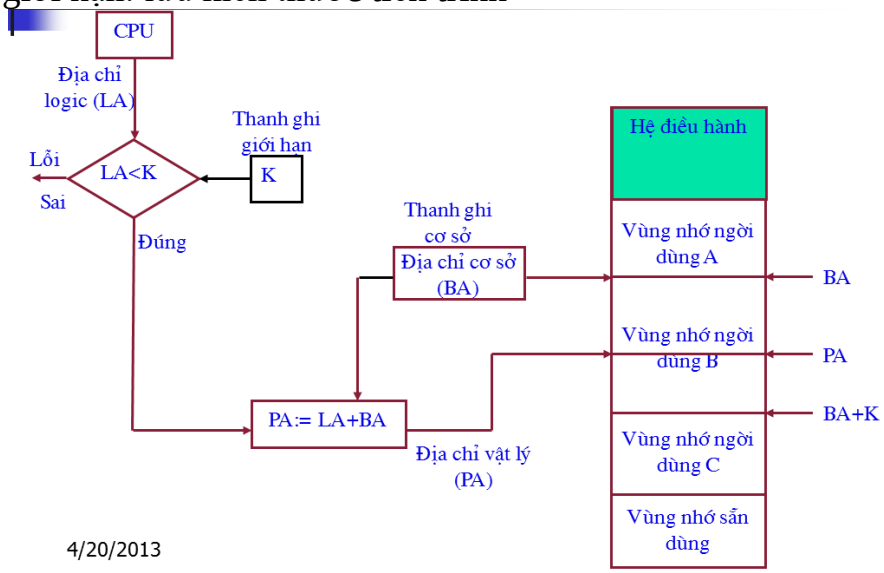
Ví dụ: Các tiến trình (trong không gian logic) có yêu cầu các phân đoạn với dung lượng ít nhất là 25856 KB, nhưng tất cả các phân đoạn trống trong bộ nhớ vật lý đều nhỏ hơn dung lượng này, do đó chúng sẽ bị lãng phí vì không thể dùng cho bất kỳ tiến trình nào.

Câu 6: Nêu biện pháp bảo vệ bộ nhớ dựa vào thanh ghi

Bổ sung vào cấu trúc phần cứng của máy tính một thanh ghi cơ sở (base register) và một thanh ghi giới hạn (limit register).

-Thanh ghi cơ sở: chứa địa chỉ bắt đầu của vùng nhớ cấp phát cho tiến trình.

-Thanh ghi giới hạn: lưu kích thước tiến trình



Mỗi địa chỉ bộ nhớ do tiến trình người dùng phát sinh ra đều so sánh với thanh ghi giới hạn, nếu nhỏ hơn nó sẽ được tự động cộng với địa chỉ chứa trong thanh ghi cơ sở để cho ra địa chỉ tuyệt đối trong bộ nhớ.

Ưu điểm:

Nhờ sử dụng thanh ghi cơ sở/ giới hạn có thể bảo vệ vùng nhớ của tiến trình người dùng.

Hỗ trợ tái định vị động nhờ thanh ghi cơ sở nên có thể di chuyển chương trình trong bộ nhớ khi chúng xử lý = thay đổi giá trị trong thanh ghi cơ sở.

Giải pháp cho việc đoạn lệnh bị ghi đè :cần tách đoạn lệnh và đoạn dữ liệu, đồng thời định rõ quyền thao tác trên các đoạn đó. Đoạn lệnh chỉ được thực hiện thao tác *chạy (execute),* đoạn dữ liệu có thể *đọc/ghi.*

Dựa vào hai cơ chế:

+ ***Hai cặp thanh ghi***:

- Mỗi đoạn lệnh và đoạn dữ liệu đều có một cặp thanh ghi biên. Thanh ghi cho đoạn lệnh đọc gán quyền chỉ đọc, thanh ghi cho đoạn dữ liệu gán quyền đọc/ghi. Như vậy, do đoạn lệnh chỉ có thể đọc nên không gây tình trạng ghi đè nữa, đồng thời đoạn lệnh này không thể bị sửa đổi.
- Nhược điểm là hạn chế đọc trên cả một đoạn lệnh

+ ***Kiến trúc gắn nhãn***

- Là một kỹ thuật bảo vệ cho mỗi từ nhớ (word), mỗi địa chỉ bộ nhớ được gắn một nhãn.
- Nhãn này có thể chứa trong 1 hay hơn 1 bit, để thiết lập các quyền thao tác có thể thực hiện được trên nội dung của địa chỉ đó, mỗi địa chỉ ta có một nhãn tương ứng.
- Việc gắn nhãn do OS thực hiện theo chế độ đặc quyền.
- Nhược điểm: Khó thực hiện, tốn công.

Câu 7: Tìm hiểu các mức bảo vệ của tiêu chuẩn DoD.

Tiêu chuẩn DoD, hệ thống an toàn có thể đợc phân theo 4 mức phân cấp (D, C, B, A). Trong mỗi mức phân cấp, lại chia thành các lớp phân cấp:

*__Mức D__ *(bảo vệ tối thiểu):* Không có lớp con nào, các hệ thống trong mức này sẽ không có bất kỳ một yêu cầu nào cần thiết để phân loại cao hơn.

***Mức C** (bảo vệ tuỳ ý)*: Các hệ thống trong mức này cung cấp các chính sách kiểm soát truy nhập tùy ý – DAC và các chính sách sử dụng lại đối tượng. Ngoài ra, chúng còn cung cấp các cơ chế nhận dạng/xác thực và kiểm toán. Mức C được chia làm hai lớp:

-*Lớp C1 (bảo vệ an toàn tùy ý)*:

+ Các hệ thống trong lớp này cung cấp các đặc trưng an toàn cho kiểm soát truy nhập tùy ý (DAC)

+ Nhận dạng/xác thực.

-*Lớp C2 (bảo vệ truy nhập có kiểm soát)*:

+ C1

+ Hệ thống phải cú khả năng lưu thông tin về người dùng đơn lẻ và có các cơ chế kiểm toán.

***Mức B** (bảo vệ bắt buộc)*: yêu cầu cơ bản với các hệ thống thuộc mức B là cần có các nhãn an toàn và chính sách kiểm soát truy nhập bắt buộc – MAC. Hầu hết các dữ liệu liên quan trong hệ thống cần phải được gán nhãn. Mức B được chia thành 3 lớp:

-*Lớp B1 (bảo vệ an toàn có gán nhãn)*:

+ C2

+ Có thêm các nhãn an toàn và chính sách kiểm soát truy nhập bắt buộc – MAC.

-*Lớp B2 (bảo vệ có cấu trúc)*:

+ B1

+ Các chính sách kiểm soát truy nhập của lớp B1 sẽ được áp dụng với tất cả chủ thể và đối tượng của hệ thống. Cả nhà quản trị và người dùng đều được cung cấp các cơ chế xác thực, và các công cụ để hỗ trợ việc quản lý cấu hình.

- *Lớp B3 (miền an toàn)*:

+ B2

+ Có khả năng chống đột nhập, các đặc tính an toàn cũng phải mạnh hơn (phục hồi, khả năng kiểm toán). Nói chung hệ thống ở lớp B3 phải có khả năng cao chống lại được các truy nhập trái phép.

***Mức A** (bảo vệ có kiểm tra)*: Đặc điểm cơ bản của lớp này là sử dụng các phương pháp hình thức để kiểm tra an toàn cho hệ thống. Mức A được chia thành:

-*Lớp A1 (thiết kế kiểm tra)*:

+ Tương đương B3

+ Tuy nhiên, hệ thống thuộc lớp A1 cần sử dụng các kỹ thuật hình thức và phi hình thức để chứng minh tính tương thích giữa đặc tả an toàn mức cao và mô hình chính sách hình thức.

-*Lớp ngoài A1 (không được mô tả).*

Chương 3 Thiết kế CSDL an toàn

Câu 1: Nêu sự khác nhau giữa hệ điều hành và hệ quản trị CSDL.

-**Độ chi tiết của đối tượng** (Object granularity): Trong OS, độ chi tiết ở mức tệp (file), thiết bị. Trong DBMS, nó chi tiết hơn (ví dụ như: các quan hệ, các hàng, các cột, các trường).

-***Các tương quan ngữ nghĩa trong dữ liệu*** *(Semantic correlations among data):* Trong OS không có, trong CSDL, dữ liệu có ngữ nghĩa và liên quan với nhau thông qua các quan hệ ngữ nghĩa như: data, time**,** context**,** history…

-***Siêu dữ liệu*** (*Metadata*): Siêu dữ liệu tồn tại trong một DBMS, cung cấp thông tin về cấu trúc của dữ liệu trong CSDL, cấu trúc lưu trữ vật lý của các đối tượng CSDL(quan hệ, thuộc tính, ràng buộc, miền…). Trong OS không có.

-***Các đối tượng logic và vật lý***: Các đối tượng trong một OS là các đối tượng vật lý (ví dụ: các file, các thiết bị, bộ nhớ và các tiến trình). Các đối tượng trong một DBMS là các đối tượng logic (ví dụ: các quan hệ, các khung nhìn) và chúng độc lập với các đối tượng của OS.

-***Nhiều loại dữ liệu***: Đặc điểm của các CSDL là có rất nhiều kiểu dữ liệu, do đó các CSDL cũng yêu cầu nhiều chế độ truy nhập (ví như chế độ thống kê, chế độ quản trị). Tại mức OS chỉ tồn tại truy nhập vật lý, bao gồm các thao tác trên file như: đọc, ghi và thực hiện.

-**Các đối tượng động và tĩnh**: Các đối tượng được OS quản lý là các đối tượng tĩnh và tương ứng với các đối tượng thực. Trong các CSDL, các đối tượng có thể

được tạo ra động (ví dụ các khung nhìn hay các kết quả hỏi đáp) và không có các đối tượng thực tương ứng.

-Các giao tác đa mức: Trong một DBMS thường có các giao tác liên quan đến dữ liệu ở các mức an toàn khác nhau (ví dụ: select, insert, update, delete), vì một đối tượng trong CSDL có thể chứa các dữ liệu ở các mức an toàn khác nhau. Tại mức OS, một đối tượng chỉ có thể chứa dữ liệu ở một mức an toàn, chỉ có các thao tác cơ bản (ví dụ, đọc, ghi, thực hiện).

-Thời gian tồn tại của dữ liệu: Dữ liệu trong một CSDL có thời gian tồn tại dài và DBMS có thể đảm bảo việc bảo vệ từ đầu đến cuối trong suốt thời gian tồn tại của dữ liệu. Nhưng dữ liệu trong một hệ điều hành thường không được lưu trữ một cách an toàn.

Câu 2: Tìm hiểu mô hình cấp quyền System R

Hệ thống R là hệ CSDL quan hệ đầu tiên của IBM năm 1970. Việc bảo vệ được thực hiện tại mức table. Có 5 chế độ truy nhập vào một table:

- **Read:** đọc các bộ của một bảng. Một user có truy nhập read, có thể định nghĩa các ***views*** trên table đó.
- **Insert**: chèn một hay nhiều bản ghi vào một table
- **Delete:** xóa các bản ghi
- **Update**: cập nhật các bản ghi
- **Drop:** xóa bảng và cấu trúc của bảng

Hệ thống R hỗ trợ quản trị quyền phi tập trung: Người tạo ra bảng có mọi đặc quyền trên bảng đó và có thể grant/revoke (trao/thu hồi) quyền cho các user khác, mỗi quyền là một bộ sau: <s, p, t, ts, g, go>

- **s**: chủ thể được gán quyền (***grantee)***.
- **p**: đặc quyền được gán (select, update…).
- **t**: tên bảng, trên đó truy nhập được gán.
- **ts**: thời điểm quyền được gán.
- **g**: người gán quyền (***grantor***).
- **go** ∈{yes,no}: ***grant option***.

Gán quyền (Grant prilvileges): Dùng câu lệnh GRANT để cấp quyền, có tùy chọn GRANT OPTION. Sự ủy quyền được thể hiện thông qua GRANT OPTION, nếu cấp quyền cho 1 user bằng câu lệnh có GRANT OPTION thì user đó ngoài việc có thể thực hiện quyền được cấp, còn có thể cấp quyền đó cho các user khác. Có thể cấp quyền (privilege) trên table và view:

GRANT privileges *ON* object *TO* users *[WITH GRANT OPTION]*

Thu hồi quyền (Revoke prilvileges): Nếu một user được gán quyền trên một table với GRANT OPTION, anh ta có thể gán và thu hồi quyền cho các user khác với các quyền anh ta có:
REVOKE [GRANT OPTION FOR] privileges *ON* object *FROM* users *{CASCADE | RESTRICT*
Mô hình quyền System R sử dụng cơ chế ***thu hồi đệ quy.*** Người dùng x thu hồi đặc quyền p trên bảng t từ người dùng y. Khi đó theo đệ quy, người dùng y sẽ thu hồi các quyền của anh ta có cho những người dùng anh ta đã gán, …tiếp tục đến khi thu hồi hết quyền. Nếu x thu hồi quyền của y, trong khi đó x không gán quyền gì cho y trước đó, thì việc thu hồi quyền này bị loại bỏ.
Khung nhìn (View): view là một cơ chế thường được dùng để hỗ trợ việc điều khiển truy cập dựa trên nội dung. Một user muốn tạo các view – khung nhìn trên các table cơ sở thì:

+ Người sở hữu table trao quyền create View.
+ User ít nhất phải có quyền read trên các bảng cơ sở này, mới có quyền tạo các view.
+ Một view có thể được tạo từ một hoặc nhiều table cơ sở (Join).
+ Người sở hữu của một khung nhìn có các quyền giống như quyền mà user đó có trên các bảng cơ sở.
+ Người sở hữu một khung nhìn (trên các bảng cơ sở, có các quyền với GRANT OPTION) thì user đó cũng có thể gán các quyền trên view cho những user khác, thậm chí những user này không có quyền truy nhập nào trên các bảng cơ sở. Sau khi tạo ra view, những quyền user bị thu hồi trên các bảng cơ sở cũng bị thu hồi trên các khung nhìn.

Câu 3: Nêu ví dụ về đặc quyền hệ thống (System Prilvilege) và đặc quyền đối tượng (Object Prilvilege), viết câu lệnh SQL cho các ví dụ đó. Nêu sự khác nhau giữa Admin option và Grant option. (Ví dụ các câu lệnh SQL).

***- Đặc quyền hệ thống*:**
Cho phép người sử dụng tạo những cơ sở dữ liệu mới, tạo các đối tượng mới bên trong cơ sở dữ liệu có sẵn, hay sao lưu cơ sở dữ liệu hoặc nhật ký giao tác.
Ví dụ một số đặc quyền hệ thống như:

- CREATE DATABASE
- CREATE TABLE
- CREATE PROCEDURE
- CREATE DEFAULT
- CREATE RULE
- CREATE VIEW
- BACKUP DATABASE
- BACKUP LOG

Ví dụ:

```
Create table SinhVien (
MaSV  varchar(10) not null primary key,
Hoten   varchar(30) not null,
GioiTinh     varchar (20),
Quequan     varchar(40),
MaLop       varchar(10),
FOREIGN KEY(MaLop) REFERENCES  Lop(MaLop))
```

- Đặc quyền đối tượng:

Các quyền dùng đối tượng cho phép người sử dụng, role thực hiện những hành động trên một đối tượng cụ thể trong cơ sở dữ liệu.

Ví dụ một số đặc quyền đối tượng:

- SELECT: Xem dữ liệu trong bảng, View, hay cột
- INSERT: Thêm dữ liệu vào bảng hoặc view.
- UPDATE: Sửa đổi dữ liệu có sẵn trong bảng, view hoặc cột.
- DELETE: Xoá dữ liệu trong bảng hoặc view
- EXECUTE: Chạy một thủ tục được lưu

REFERENCE: Tham khảo một bảng bằng khoá ngoại

Ví dụ:

*Select * from SinhVien where MaLop = 'ML01'*

Update SinhVien Set Hoten = 'Nguyen Thi Minh' where MaSV = 'MS17'

**** Sự khác nhau giữa Admin option và Grant option:***

- Admin option là tùy chọn trong câu lệnh gán quyền hệ thống, cho phép chủ thể lan truyền quyền đó cho chủ thể khác.
- GrantAdmin option là tùy chọn trong câu lệnh gán quyền đối tượng, cho phép chủ thể lan truyền quyền đó cho chủ thể khác.

Câu 4: Điều khiển kiểm soát truy nhập phụ thuộc:

-Các kiểm soát phụ thuộc tên (*Name-dependent controls*) dựa vào tên của đối tượng bị truy nhập.

-Các kiểm soát phụ thuộc dữ liệu (*Data-dependent controls*) thực hiện truy nhập phụ thuộc vào các nội dung của đối tượng bị truy nhập.

-Các kiểm soát phụ thuộc ngữ cảnh (*Context-dependent controls*) chấp thuận hoặc từ chối truy nhập phụ thuộc vào giá trị của một số biến hệ thống (ví dụ như: ngày, tháng, thiết bị đầu cuối yêu cầu – vị trí người sử dụng).

-Các kiểm soát phụ thuộc lược sử (*History-dependent controls*) quan tâm đến các thông tin về chuỗi câu truy vấn (ví dụ như: các kiểu câu truy vấn, dữ liệu trả lại, profile của người dùng đang yêu cầu, tần suất yêu cầu).
-Các kiểm soát phụ thuộc kết quả (*Result-dependent controls*) thực hiện quyết định truy nhập phụ thuộc vào kết quả của các thủ tục kiểm soát hỗ trợ, chúng là các thủ tục được thực hiện tại thời điểm hỏi.

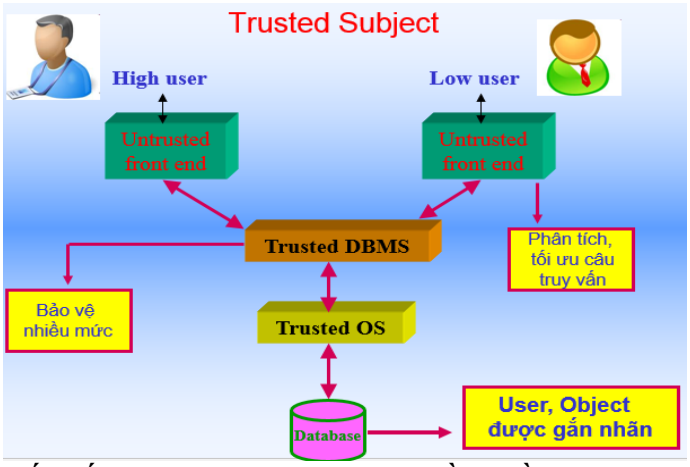
Câu 5: Tìm hiểu đặc điểm cơ bản của kiến trúc chủ thể tin cậy (Trusted Subject) và kiến trúc Woods Hole. Mô tả chi tiết 3 kiến trúc Woods Hole là: Integrity Lock, Kernelized, Replicated, 3 kiến trúc này có trong những sản phẩm thương mại nào?

***Kiến trúc chủ thể tin cậy Trusted Subject**

Đặc điểm:

- Giả thiết DBMS và một OS tin cậy.
- DBMS hoạt động như là một chủ thể tin cậy của OS
- DBMS có trách nhiệm trong việc bảo vệ đa mức (multilevel) các đối tượng của CSDL.

- Được sử dụng trong nhiều DBMS thương mại (Sybase, Informix, Ingres, Oracle, DEC, Rubix).

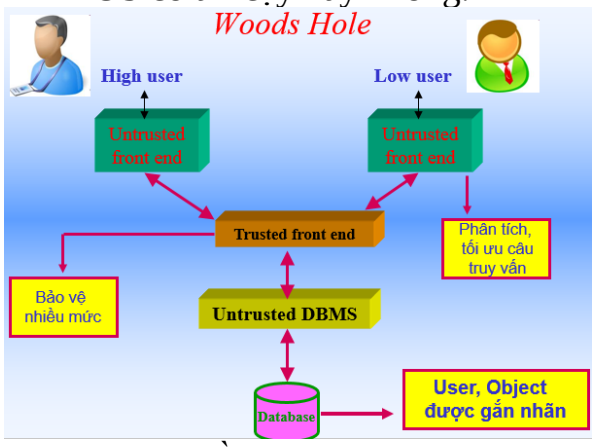


- Người dùng kết nối tới DBMS qua các phần mềm *untrusted front end* (vì họ kết nối qua Internet).
- Người dùng được phân loại các mức nhạy cảm khác nhau: High (cao), Low (thấp), và một mức DBMS khác với hai mức trên.
- Các chủ thể và đối tượng được gán một nhãn DBMS không giống với mức High và Low.
- Chỉ có các chủ thể được gán nhãn DBMS mới được phép thực hiện mã lệnh và truy nhập vào dữ liệu.
- Các chủ thể có nhãn DBMS được coi là các chủ thể tin cậy và được miễn kiểm soát bắt buộc của OS
- Các đối tượng CSDL được gán nhãn nhạy cảm (ví dụ: các bộ, các giá trị).

- Hệ quản trị Sybase tuân theo giải pháp này, với kiến trúc máy khách/máy chủ, Sybase thực hiện gán nhãn mức bản ghi (mức hàng).

*** Kiến trúc Woods Hole**

Các kiến trúc Woods Hole sử dụng DBMS không tin cậy cùng với một bộ lọc tin cậy và không quan tâm đến OS có tin cậy hay không.



-Phần mềm front ends và DBMS đều không tin cậy (Không quan tâm OS có tin cậy hay không)

-Phần mềm untrusted front-end thực hiện các công việc xử lý trước và sau các câu truy vấn (phân tích, tối ưu hóa, phép chiếu).

-Phần mềm trusted front end (TFE) ở giữa thực thi các chức năng an toàn và bảo vệ nhiều mức, vì vậy hoạt động như một TCB (Trusted Computing Base).

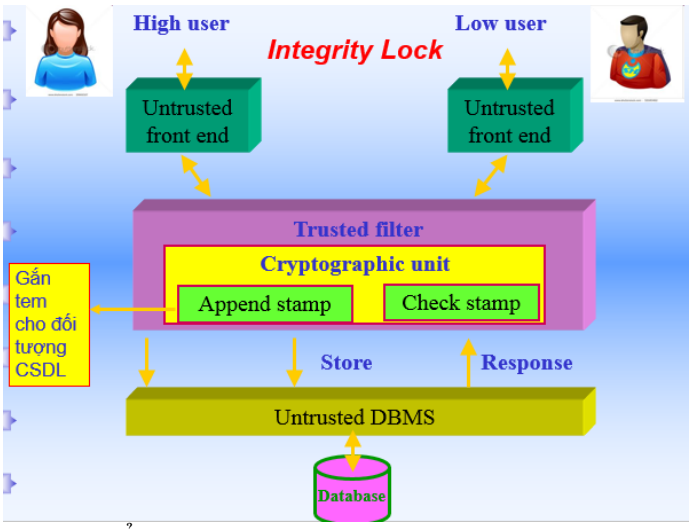
***Kiến trúc Integrity Lock**

-*Khoá toàn vẹn* được đề xuất lần đầu tiên tại Viện nghiên cứu của Lực lượng Không quân về An toàn cơ sở dữ liệu [AF83], được dùng để kiểm soát *tính toàn vẹn* và sự *truy nhập* cho cơ sở dữ liệu.

-Kiến trúc Integrity lock đã có trong hệ quản trị thương mại TRUDATA.

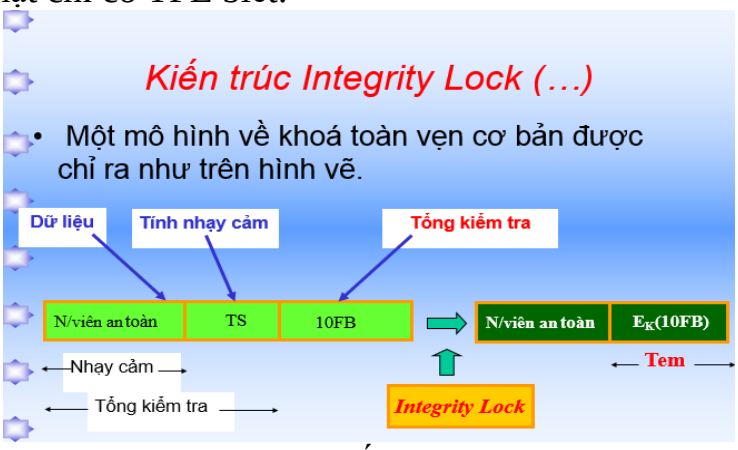
Đặc điểm:

- TFE thực thi bảo vệ nhiều mức bằng cách gắn các nhãn an toàn vào các đối tượng CSDL dưới dạng các *tem – Stamps.*
- Một *tem* là một trường đặc biệt của một đối tượng, lưu thông tin về nhãn an toàn và các dữ liệu điều khiển liên quan khác.
- *Tem* là dạng mã hóa của các thông tin trên, sử dụng một kỹ thuật niêm phong mật mã gọi là **Integrity Lock**.



TFE có nhiệm vụ tạo và kiểm tra các tem.

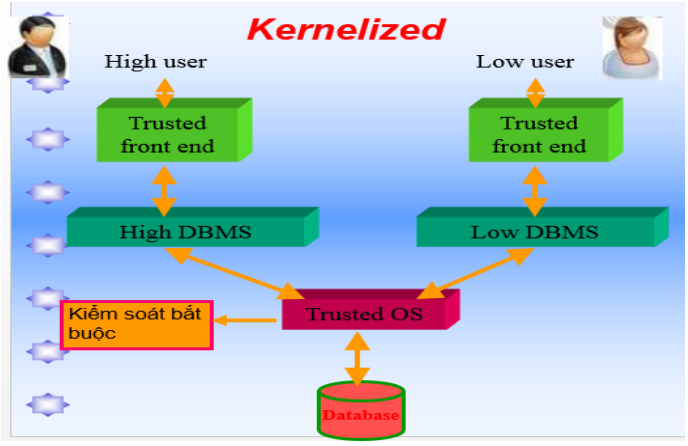
- TFE sử dụng mật mã khóa bí mật để tạo *tem* và giải mã các tem. Các tem này có thể tạo ra dựa vào tổng kiểm tra (checksum).
- Khóa bí mật chỉ có TFE biết.



- ***Insert dữ liệu****:* khi người dùng muốn insert một mục dữ liệu, TFE sẽ tính:
  - *Tổng kiểm tra = mức nhạy cảm dữ liệu + dữ liệu.*
  - *Mã hoá* tổng kiểm tra này bằng một khoá bí mật K, tạo ra *tem*, và lưu vào trong CSDL cùng với mục dữ liệu đó (gắn với mục dữ liệu).
- ***Đưa ra dữ liệu:*** Khi đưa ra dữ liệu trả cho người dùng, TFE nhận được dữ liệu từ DBMS không tin cậy, nó sẽ kiểm tra tem gắn với mục dữ liệu xem có chính xác không:
  - Giải mã tem gắn với dữ liệu.
  - So sánh dữ liệu nhận được với dữ liệu sau khi giải mã tem. Nếu không khớp chứng tỏ dữ liệu đã bị sửa đổi.
  - ***Lưu ý:*** nếu dùng hàm băm để tạo tem, thì sau khi DBMS nhận được dữ liệu và tem tương ứng, nó sẽ băm dữ liệu này ra và so sánh với tem nhận được xem có trung nhau không.

***Kiến trúc Kernelized**

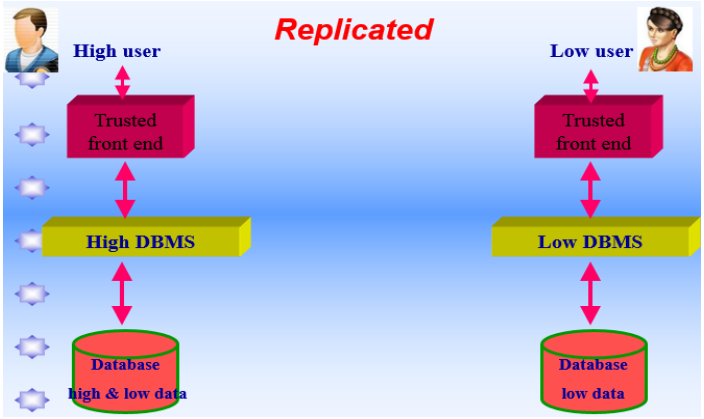
- Sử dụng một OS tin cậy, có trách nhiệm đối với các truy nhập vật lý vào dữ liệu (trong CSDL) và có trách nhiệm tuân theo bảo vệ bắt buộc.
- High User (người dùng làm việc ở mức cao) tương tác với một High DBMS, thông qua một TFE, Low User (người dùng làm việc ở mức thấp) tương tác với một Low DBMS cũng thông qua một TFE.
- Sau đó, các yêu cầu của họ được chuyển cho OS, và OS lấy lại dữ liệu hợp lệ từ CSDL.
- Đã có trong mẫu thử Sea View và trong hệ quản trị thương mại Oracle.
- ***Quá trình khôi phục***: được thực hiện khi OS cần lấy lại dữ liệu được lưu giữ trên các đối tượng của nó để trả về cho người dùng. Để từ các quan hệ đơn mức, sinh ra một khung nhìn đa mức chỉ chứa các dữ liệu mà người dùng yêu cầu,
- ***Quá trình phân tách*:** thực hiện chuyển đổi một quan hệ đa mức (đối tượng CSDL) thành một số quan hệ đơn mức, (chỉ chứa dữ liệu ở một mức an toàn nào đó), được lưu giữ trong các đối tượng OS.
- Các đối tượng (có các nhãn an toàn giống nhau) của CSDL được lưu giữ trong các đối tượng của OS tin cậy. Vì vậy, OS tin cậy tiến hành kiểm soát an toàn trên các đối tượng lưu giữ này, cần có các *quá trình phân tách* và *khôi phục* quan hệ nhiều mức.



***Kiến trúc Replicated**

- Có trong mẫu thử NRL, nhưng chưa có trong DBMS thương mại nào, vì nó rất đắt!
- Người dùng mức cao có thể xem và sửa đổi cả dữ liệu mức thấp và mức cao.
- Để tuân theo giải pháp này cần có các thuật toán đồng bộ an toàn để đảm bảo tính tương thích lặp và chi phí lặp cũng rất lớn.
- *Dữ liệu mức thấp được lặp trong CSDL.*

Người dùng mức thấp chỉ được phép truy nhập vào CSDL độ ưu tiên thấp, không có khả năng sửa đổi dữ liệu mức cao.



Câu 6: Các bước thiết kế một cơ sở dữ liệu an toàn. Yêu cầu: khi cần thiết kế 1 CSDL an toàn phải đưa ra được các giải pháp an toàn cho bài toán đó

Có 5 bước thiết kế một CSDL an toàn:

Bước 1. Phân tích sơ bộ

+ Đánh giá các rủi ro
+ Ước lượng các chi phí thiết kế
+ Phát triển các ứng dụng cụ thể nào

Bước 2. Các yêu cầu và các chính sách an toàn

- *Phân tích yêu cầu:*
  + Phân tích giá trị : xác định mức nhạy cảm của dữ liệu.
  + Nhận dạng đe doạ/phân tích điểm yếu
  +Phân tích và đánh giá rủi ro: khả năng xảy ra của các biến cố không mong muốn và tác động của chúng.
  + Xác định yêu cầu bảo vệ
- *Lựa chọn chính sách:*
  + Chính sách là các quy tắc ở mức cao, bắt buộc phải tuân theo trong các quá trình thiết kế, thực thi và quản lý hệ thống an toàn.
  + Định nghĩa các chế độ truy nhập (đọc, ghi) của chủ thể vào các đối tượng của hệ thống

Bước 3. Thiết kế khái niệm

- Xây dựng mô hình E-R: Vẽ các thực thể và mối quan hệ giữa các thực thể đó.

Bước 4. Thiết kế lôgíc

- Xây dựng các lược đồ cơ sở dữ liệu (là các bảng và các thuộc tính của chúng)

Bước 5. Thiết kế vậy lý

- Thiết kế thật sự vào hệ thống

Chương 4 Cơ sở dữ liệu thống kê

Câu 1: Cơ sở dữ liệu thống kê (statistical database) là gì? (Viết được các câu lệnh SQL cho các thống kê). Ứng dụng trong thực tế?

Cơ sở dữ liệu thống kê (SDB)- Statistical database

+ Là một CSDL được sử dụng cho mục đích phân tích thống kê.

+ Là một CSDL chứa các bản ghi nhạy cảm mô tả về các cá nhân nhưng chỉ các câu truy vấn thống kê (như: **COUNT, SUM, AVERAGE, MAX, MIN…)** mới được trả lời, ngoài các câu truy vấn này thì những truy vấn vào các mục dữ liệu riêng sẽ không được đáp lại

- Sự khác biệt chính với cơ sở dữ liệu quan hệ thông thường đó là với một SDB những câu truy vấn thống kê mới được phép truy vấn, những câu truy vấn vào từng trường dữ liệu riêng lẻ đều bị coi là không hợp lệ.

- Ngoài ra, cơ sở dữ liệu quan hệ được lưu dưới dạng bảng nhưng SDB có hai loại mô tả, một là dạng quan hệ, hai là dạng vĩ mô cũng là các bảng nhưng là các bảng thống kê.

Câu 2: Công thức đặc trưng

Công thức đặc trưng là một công thức lôgíc, được ký hiệu bởi một chữ cái viết hoa (*A,B,C,...*), trong đó các giá trị thuộc tính được kết hợp với nhau thông qua các toán tử Boolean như OR, AND, NOT ($\vee,\wedge,\neg$).

Ví dụ:

C = (GioiTinh=F)$\wedge$ [(MaPhong="Kế hoạch")$\vee$ (MaPhong="Tài vụ")] $\wedge$(NamSinh<1965)

Câu 3: Thế nào là thống kê nhạy cảm, cho ví dụ? Working knowledge và Supplementary knowledge?

**** Thống kê nhạy cảm trong 1 cơ sở dữ liệu thống kê:***

Thống kê được tính toán trên một *thuộc tính bí mật* trong tập truy vấn có kích cỡ bằng 1 là thống kê nhạy cảm.

- *Ví dụ:* Với công thức đặc trưng *C = (AGE>50)*

Ta có, *COUNT(AGE >50) =1*

Khi đó mọi thống kê trên *C* đều là thống kê nhạy cảm, chẳng hạn:

SUM(Salary, age>50) là thống kê nhạy cảm

**** Working knowledge và Supplementary knowledge***:

- *Kiến thức làm việc (working knowledge):* Là tập các mục thông tin (field) và giá trị thuộc tính trong SDB và các kiểu thống kê có sẵn trong SDB mà người dùng có thể biết một cách hợp lệ.
- *Kiến thức bổ sung của người sử dụng (supplementary knowledge):* Người sử dụng có thể có kiến thức bên ngoài về các cá nhân được biểu diễn trong SDB. Người dùng hoàn toàn có thể lợi dụng kiến thức này cho các mục đích xấu để suy diễn các thông tin nhạy cảm.

Câu 4: Nêu cách thức tấn công trực tiếp. Nêu ví dụ.

-Sử dụng các câu truy vấn thông thường, không phải truy vấn thống kê

VD: ***SELECT Ten FROM NhanVien WHERE Luong>4.360***

Giải pháp: Bộ lọc - Filter (loại các truy vấn không hợp lệ)

Câu 5: Tấn công dựa vào đếm

-Đây là loại tấn công bằng cách kết hợp giá trị đếm với giá trị tổng để thu được thông tin bí mật.

VD: ***COUNT (ChucVu = "Trưởng phòng", Phong= "Kế hoạch") = 1***

=> SUM (Luong, (ChucVu= "Trưởng phòng", Phong= "Kế hoạch"))

1. Nêu cách thức tấn công của dựa vào Trình theo dõi. Nêu ví dụ.
2. Nêu cách thức tấn công dựa vào Hệ tuyến tính. Nêu ví dụ.
3. Tìm hiểu các kỹ thuật chống suy diễn trong CSDL thống kê, nêu ưu nhược điểm của từng phương pháp. (Chú ý tìm hiểu kỹ các kiểm soát này)

Kiểm soát kích cỡ tập truy vấn

Kỹ thuật gây nhiễu

Kiểm soát dựa vào hạn chế

Câu 6: Nêu cách thức tấn công của dựa vào Trình theo dõi. Nêu ví dụ.

****Trình theo dõi (Tracker):***

- Là một tập các công thức đặc trưng, có thể được sử dụng để đưa thêm bản ghi vào các các tập truy vấn kích cỡ nhỏ, làm cho kích cỡ của chúng nằm trong khoảng **[*k*, *N*-*k*].**
- Thông qua các trình theo dõi có thể tính toán được các thống kê bị hạn chế.

- Giả sử **C** là công thức đặc trưng người dùng yêu cầu
- **T** là một trình theo dõi. T thỏa mãn điều kiện:

K<=|X(T)|<=N-K

Kiểu 1

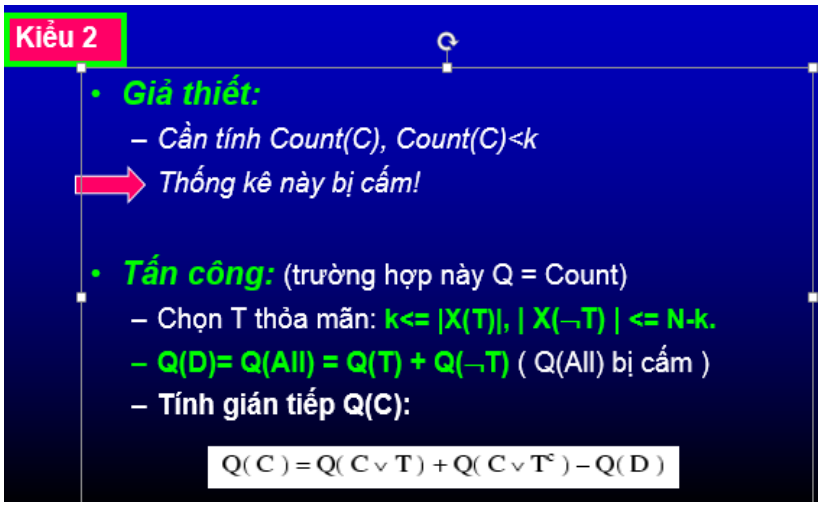
K=2

- ***Giả thiết:***
  - User cần tính **Count(C), Sum(C, Luong)**
  - Công thức ***C = (A∧B),*** và **Count (C) = 1**.

  Câu truy vấn này bị cấm!
- ***Tấn công:***
  - Tính **T = *A*∧¬ *B*** thỏa mãn **k<=|X(T)|<= N-k.**
  - Tính gián tiếp Count (C):

***Q*(C)= *Q* (A∧B) = *Q* (A)- *Q* (A∧¬B)**

=> *Q* (C) = *Q* (A) - *Q* (T)



Câu 7: Nêu cách thức tấn công dựa vào Hệ tuyến tính. Nêu ví dụ.

- Là loại tấn công bằng cách giải một hệ phương trình có dạng: $HX = Q$

$$\lambda_{1,1}x_1 + \lambda_{1,2}x_2 + \ldots + \lambda_{1,n}x_N = q_1$$
$$\lambda_{2,1}x_1 + \lambda_{2,2}x_2 + \ldots + \lambda_{2,N}x_N = q_2$$
$$\vdots$$
$$\lambda_{k,1}x_1 + \lambda_{k,2}x_2 + \ldots + \lambda_{k,n}x_N = q_K$$

Mỗi phương trình tương ứng một câu truy vấn

- H là ma trận truy vấn
  - $H[i,j] = 1$ nếu bản ghi $x_j \in X(C_i)$, (tương ứng q_i)
  - $H[i,j] = 0$ nếu ngược lại

$$H = \begin{vmatrix} \lambda_{1,1} & \lambda_{1,2} & \ldots & \lambda_{1,n} \\ \lambda_{2,1} & \lambda_{2,2} & \ldots & \lambda_{2,n} \\ \vdots & \vdots & & \vdots \\ \lambda_{k,1} & \lambda_{k,2} & \ldots & \lambda_{k,n} \end{vmatrix}$$

  - $x_1, \ldots, x_N$ là giá trị của N bản ghi
- $Q = (q_1, \ldots, q_k)$ là vector của các thống kê đưa ra

- ***C = (Phong='Kế hoạch') ∧(GioiTinh=F)***

- Cần tính q= Sum(Luong, C)
- Tính q_1=X(C_1) = Count(Phong=‘Kế hoạch’) = 3
- Tính q_2 =X(C_2) = Count(Phong=‘Kế hoạch’, GioiTinh =M)=2
- Sum(Luong, C) = Sum(Luong, C_1) – Sum(Luong,C_2)
  = (6200+4000+2900) – (6200+4000) = 2900.
- Như vậy, kẻ tấn công đã tìm ra lương của người thỏa mãn C.

Câu 8 : Tìm hiểu các kỹ thuật chống suy diễn trong CSDL thống kê, nêu ưu nhược điểm của từng phương pháp. (Chú ý tìm hiểu kỹ các kiểm soát này)

***Kiểm soát kích cỡ tập truy vấn**

Một thống kê *q*(C) chỉ được phép nếu tập truy vấn của nó, *X(C)*, thoả mãn quan hệ sau:

$$k \le |X(C)| \le N-k$$

$$0 \le k \le N/2$$

Trong đó, *N* là tổng số bản ghi trong SDB, *k* do DBA định nghĩa.

Kiểm soát này ngăn chặn các tấn công đơn giản, dựa vào các tập truy vấn rất nhỏ hoặc rất lớn.

Ưu điểm:

- Đưa ra kết quả chính xác
- Chỉ chống được tấn công suy diễn đơn giản

Nhược điểm:

- Không chống được một số tấn công phức tạp như: Trình theo dõi, Hệ tuyến tính.
- Hạn chế khả năng hữu ích của SDB (vì hạn chế nhiều câu truy vấn)

Ví dụ: Người dùng yêu cầu thống kê *q*1 = *Count* (C) =1, => có một cá nhân A thỏa mãn C.

Đưa ra thống kê *q*2 = *Count* (C $\wedge$ C')

- Nếu $q_2 = 1$ => *A* thỏa mãn C'
- Ngược lại, A không thỏa mãn C'

Đưa ra thống kê khác, ví dụ Sum(C, A_i)

=> Kiểm soát kích cỡ tập truy vấn không cho phép đưa ra q_1, q_2.

***Kiểm soát kích cỡ tập truy vấn mở rộng:**

-Nhược điểm của kiểm soát kích cỡ tập truy vấn là do các công thức đặc trưng liên quan đến nhau (ví dụ: C và T).

-*Cải tiến*: tăng số lượng các tập truy vấn cần được kiểm soát.

- Cho công thức đặc trưng C
- *Tìm tập truy vấn ngầm định* của C

-Cho trước một thống kê bậc *m* có dạng như sau:

- $q(A_1 = a_1 \wedge A_2 = a_2 \wedge \ldots \wedge A_m = a_m)$ Hoặc:
- $q(A_1 = a_1 \vee A_2 = a_2 \vee \ldots \vee A_m = a_m)$

-Khi đó, tồn tại $2^m = C_m^0 + C_m^1 + C_m^2 + \ldots + C_m^{m-1}$ tập truy vấn ngầm định, tương ứng với các thống kê sau đây:

$q(A_1 = a_1 \wedge A_2 = a_2 \wedge \ldots \wedge A_m = a_m)$

$q(A_1 = a_1 \wedge A_2 = a_2 \wedge \ldots \wedge \neg A_m = a_m)$

...

$q(A_1 = a_1 \wedge \neg A_2 = a_2 \wedge \ldots \wedge A_m = a_m)$

$q(\neg A_1 = a_1 \wedge A_2 = a_2 \wedge \ldots \wedge A_m = a_m)$

...

$q(\neg A_1 = a_1 \vee \neg A_2 = a_2 \vee \ldots \vee \neg A_m = a_m)$

Ưu điểm:

- Chống được các kiểu tấn công: Trình theo dõi, Hệ tuyến tính

Nhược điểm:

- Tốn công: phải kiểm tra 2^m tập truy vấn ngầm định (hàm mũ tăng rất lớn theo *m*).

⇨ *Giải pháp này khó thực hiện*

- Ngoài tập truy vấn ngầm định, kẻ tấn công có thể sử dụng những công thức khác liên quan đến tập truy vấn này để tính ra truy vấn yêu cầu.

***Kỹ thuật gộp**

-Các câu truy vấn thống kê được tính toán trên các cá thể tổng hợp. Dữ liệu riêng sẽ được nhóm lại thành một khối nhỏ trước khi đưa ra.

-*Giá trị trung bình* của nhóm gộp sẽ thay thế cho mỗi giá trị riêng của dữ liệu được gộp

-Kỹ thuật này giúp ngăn chặn khám phá dữ liệu riêng.

Ưu điểm:

- Tránh được việc để lộ thông tin nhạy cảm

Nhược điểm:

- Kết quả đưa ra không chính xác

Ví dụ: Cục thống kê nông nghiệp quốc gia (NASS) công bố dữ liệu về các nông trường, trang trại. Để bảo vệ chống lại sự khám phá dữ liệu, dữ liệu chỉ được đưa ra ở ***mức vùng***. Dữ liệu tại các nông trại ở mỗi vùng sẽ được gộp để bảo vệ tính riêng tư và tránh bị khám phá.

***Kỹ thuật giấu ô**

-*Giấu ô*: trong các bảng, giấu đi tất cả các ô tương ứng với các thống kê nhạy cảm và các ô tương ứng với các thống kê có thể gián tiếp khám phá ra các thống kê nhạy cảm (*Giấu bổ sung)*.

-*Tiêu chuẩn giấu ô:*

- *Thống kê Count:* kích cỡ tập truy vấn bằng 1, nghĩa là Count(C) =1
- Thống kê *Sum*, tiêu chuẩn nhạy cảm được sử dụng là quy tắc *«đáp ứng n, trội k% »* . Theo tiêu chuẩn này, một thống kê là nhạy cảm nếu *n* giá trị thuộc tính của *n* hoặc ít hơn *n* bản ghi tạo thành *k*% hoặc lớn hơn *k%* trong toàn bộ thống kê *Sum* đó. Các tham số *n* và *k* được giữ bí mật và do DBA xác định.

Ưu điểm: Chống được các tấn công kết hợp dựa vào Count và Sum

Nhược điểm: Hạn chế khả năng hữu ích của SDB, vì phải che giấu một số ô trong CSDL.

***Kỹ thuật gây nhiễu**

- ***Gây nhiễu cố định (fixed perturbation)***
  - ***Ưu điểm:***
    - Chống được nhiều tấn công, kể cả tấn công tính trung bình (lặp nhiều lần)
  - ***Nhược điểm:***
    - Chỉ áp dụng cho thuộc tính số
    - Kết quả trả về không chính xác
- ***Gây nhiễu dựa vào truy vấn***
- ***Ưu điểm:***

  - Gây nhiễu dữ liệu nên chống được nhiều tấn công
- ***Nhược điểm:***
  - Với mỗi thống kê, lại phải áp dụng một hàm gây nhiễu **f**, với gía trị nhiễu=> tốn công, giảm hiệu năng hệ thống.
  - Kết quả đưa ra không chính xác.

Chương 5 Phát hiện xâm nhập trái phép

Câu 1: Tại sao phải bảo vệ CSDL? Phương pháp mã hóa CSDL cần giải quyết những vấn đề gì? Đưa ra nhận xét so với các phương pháp bảo vệ khác.

*** Tại sao phải bảo mật CSDL?**

- Một CSDL cung cấp những thông tin quan trọng của khách hàng, kế hoạch phát triển của một doanh nghiệp, các dự đoán kinh tế, và nhiều mục đích quan trọng khác…
- Sẽ có lợi cho một tin tặc khi tấn công vào CSDL hơn là nghe nén giao tiếp trên mạng.
- Dữ liệu thường được mã hóa trên đường truyền nhưng lại lưu dưới dạng rõ trong CSDL.
- Sự cố về an ninh xảy ra với CSDL có thể ảnh hưởng nghiêm trọng đến danh tiếng của công ty và quan hệ với khách hàng.

***Phương pháp mã hóa cần giải quyết vấn đề :**

Giải pháp đơn giản nhất bảo vệ dữ liệu trong CSDL ở mức độ tập tin, chống lại sự truy cập trái phép vào các tập tin CSDL là hình thức mã hóa. Tuy nhiên, mã hóa dữ liệu ở mức độ này là giải pháp mang tính "được ăn cả, ngã về không", giải pháp này không cung cấp mức độ bảo mật truy cập đến CSDL ở mức độ bảng (table), cột (column) và dòng (row). Một điểm yếu nữa của giải pháp này là bất cứ ai với quyền truy xuất CSDL đều có thể truy cập vào tất cả dữ liệu trong CSDL. Điều này phát sinh một nguy cơ nghiêm trọng, cho phép các đối tượng với quyền quản trị (admin) truy cập tất cả các dữ liệu nhạy cảm. Thêm vào đó, giải pháp này bị hạn chế vì không cho phép phân quyền khác nhau cho người sử dụng CSDL.

*** Nhận xét so với các phương pháp bảo vệ khác:**

- Rủi ro lớn nhất có lẽ là mất các khóa -> Dẫn đến mất toàn bộ dữ liệu

- Quá trình sinh khóa không đủ ngẫu nhiên -> Dẫn đến có thể ‘dễ dàng đoán’ các khóa
- Thực thi mã hóa gặp sự cố ->Dẫn đến mã hóa ‘tồi’

Câu 2: Định nghĩa hệ thống phát hiện xâm nhập (IDS). So sánh với hệ thống ngăn chặn xâm nhập (IPS)

IDS là hệ thống phần mềm hoặc phần cứng chuyên dụng tự động thực hiện quy trình giám sát các sự kiện trong mạng, thực hiện phân tích để phát hiện những vấn đề an ninh cho hệ thống.

- Ưu điểm:

+ Có khả năng phát hiện các cuộc tấn công, xâm nhập từ bên trong cũng như bên ngoài hệ thống.

+ Những thông tin hệ thống IDS cung cấp sẽ giúp chúng ta xác định phương thức, và kiểu loại tấn công, xâm nhập => đưa ra phương án phòng chống.

- Nhược điểm:

+ IDS là một hệ thống giám sát thụ động, cơ chế ngăn chặn các cuộc tấn công, xâm nhập trái phép rất hạn chế ***(Không chống được tấn công).***

+ Phần lớn, hệ thống IDS sẽ đưa ra các cảnh báo khi các cuộc tấn công, xâm nhập đã ảnh hưởng tới hệ thống rồi!

Sự khác nhau giữa IPS, IDS:

- IDS chỉ có thể phát hiện tấn công và gửi cảnh báo còn IPS là hệ thống kết hợp IDS với tường lửa, do đó nó vừa phát hiện vừa có thể chống lại tấn công và gửi cảnh báo cho nhà quản trị.

Ví dụ: Snort, Argus, một số sản phẩm của ISS như Internet Scanner, ...

Câu 3: So sánh hệ thống IDS trên máy trạm (HIDS) và hệ thống IDS trên mạng (NIDS).

- IDS là hệ thống phần mềm hoặc phần cứng chuyên dụng tự động thực hiện quy trình giám sát các sự kiện trong mạng, thực hiện phân tích để phát hiện những vấn đề an ninh cho hệ thống.
- Ví dụ: Snort, Argus, Internet Scanner…

HIDS

- Triển khai trên các máy riêng lẻ
- Phụ thuộc hệ điều hành
- Bảo vệ tài nguyên trên một máy trạm, phạm vi hẹp
- Dễ cài đặt

NIDS

- Triển khai trên một hoặc nhiều máy khác nhau
- Không phụ thuộc hệ điều hành
- Phạm vi rộng, cho toàn hệ thống mạng
- Khó cài đặt

Ví dụ: các sản phẩm của ISS

Câu 4 : Trình bày 2 mô hình phát hiện xâm nhập trong hệ thống IDS (phát hiện sự lạm dụng và phát hiện trình trạng bất thường)? Nêu ưu, nhược điểm của từng mô hình. Cho ví dụ.

* Có 2 phương pháp phát hiện xâm nhập trong hệ thống IDS là

- *Phát hiện sự lạm dụng* (*Misuse detection models*):

+ Phân tích các hoạt động của hệ thống, tìm kiếm các sự kiện giống với các ***mẫu tấn công*** đã biết trước.

+ ***Ưu điểm:***phát hiện các cuộc tấn công nhanh và chính xác, không đưa ra các cảnh báo sai làm giảm khả năng hoạt động của mạng và giúp các người quản trị xác định các lỗ hổng bảo mật trong hệ thống của mình.

+ ***Nhược điểm:*** là không phát hiện được các cuộc tấn công không có trong cơ sở dữ liệu, các kiểu tấn công mới, do vậy hệ thống luôn phải cập nhật các mẫu tấn công mới.

- *Phát hiện tình trạng bất thường* (*Anomaly detection models*):

+ Ban đầu, chúng lưu giữ các mô tả sơ lược về các hoạt động bình thường của hệ thống.

+ Các cuộc tấn công xâm nhập gây ra các hoạt động bất bình thường và kỹ thuật này phát hiện ra các hoạt động bất bình thường đó.

- Phát hiện dựa trên mức ngưỡng,
- Phát hiện nhờ quá trình tự học,
- Phát hiện dựa trên những bất thường về giao thức)

+ ***Ưu điểm:*** có thể phát hiện ra các kiểu tấn công mới, cung cấp các thông tin hữu ích bổ sung cho phương pháp dò sự lạm dụng

+ ***Nhược điểm:*** thường tạo ra một số lượng các cảnh báo sai làm giảm hiệu suất hoạt động của mạng.

- Ứng dụng của 2 phương pháp phát hiện xâm nhập này: phương pháp dựa trên mẫu được hầu hết các IDS sử dụng còn phương pháp phát hiện tình trạng bất thường chỉ dùng cho các IDS thông minh, và không nhiều các IDS hiện nay được tích hợp phương pháp này.

Câu 5: Trình bày về các tấn công vào CSDL

- ***Tấn công bên trong:*** tin tặc là người bên trong tổ chức (bên trong firewall), biết về kiến trúc của mạng.
- ***Tấn công bên ngoài***: tin tặc phải vượt qua firewall, IDS và không biết về kiến trúc của mạng

*Tấn công bí mật:

- ***Định nghĩa:*** Là loại tấn công trong đó, những người dùng bất hợp pháp có khả năng truy nhập vào thông tin nhạy cảm của CSDL.
  - Kiểm soát mức thấp nhất là đọc CSDL.
- ***Ví dụ:*** tin tặc có thể kiểm soát toàn bộ máy chủ CSDL, do đó anh ta có thể
  - Download toàn bộ file CSDL.
  - Nạp file vào Database engine để truy nhập dữ liệu như người dùng bình thường.
- ***Kiểm soát truy nhập:*** thường được sử dụng để bảo vệ CSDL, nhưng chưa đủ!
  - Thường được cấu hình chưa đúng
  - Tạo khe hở (backdoor) cho những người dùng muốn lạm dùng quyền.
- ***Việc backup CSDL không an toàn:*** là một khả năng cho kẻ tấn công có thể truy nhập vào dữ liệu nhạy cảm.
- ***Tấn công SQL Injection:*** do người lập trình yếu, tạo khe hở để kẻ tấn công truy nhập trái phép CSDL (thường trong các ứng dụng Web).
- ***Truy nhập vào file CSDL vật lý***
- ***Giải pháp:***
  - Mã hóa file CSDL, mã hóa CSDL (các bảng, khung nhìn…những thông tin bí mật).
  - Áp dụng các cơ chế bảo vệ mức cao cho bản thân CSDL (như: dùng Label - Multilevel)

*Tấn công tính toàn vẹn

- ***Định nghĩa:*** Là loại tấn công gây ra những sửa đổi trái phép đối với thông tin trong CSDL.
  - Yêu cầu: kẻ tấn công phải có khả năng Write CSDL.
- ***Một số tấn công tính toàn vẹn phổ biến:***
  - Tấn công từ các admin ác ý
  - Sự gây hại của các ứng dụng bị lỗi
  - Sử dụng tài khoản đánh cắp có truy nhập write CSDL.
  - Khả năng leo thang đặc quyền của một số tài khoản (escalating privileges)
- ***Giải pháp:***
  - ***Tách bạch nhiệm vụ (Separaton of duties):***

- Nguyên tắc này được đưa ra nhằm hạn chế tối đa một cá nhân bất kỳ có thể phá hoại dữ liệu, để đảm bảo toàn vẹn dữ liệu.
- Tách bạch nhiệm vụ được gắn liền với các kiểm soát trên các chuỗi giao tác. Để chuỗi này hoàn thành phải có nhiều hơn một người tham gia (Ví dụ giao dịch ngân hàng).

– Chỉ những ***người dùng hợp pháp*** mới được phép thực hiện những ***ứng dụng (đã được phê duyệt)*** để thay đổi thông tin trong CSDL.