**Câu 1:** **Yêu cầu cơ bản an toàn đối với hệ điều hành**

Yêu cầu cơ bản của an toàn với hệ điều hành là: Đảm bảo các chương trình hoạt động một cách an toàn. Nó phụ thuộc vào việc triển khai đúng đắn các cơ chế chia sẻ (truy nhập) và điều độ tài nguyên:  
+ Các cơ chế truy nhập tài nguyên phải xác định ranh giới giữa các tài nguyên và đảm vảo các thao tác tới các tài nguyên không bị xung đột  
+Cơ chế điều độ đảm bảo tính sẵn sàng của tài nguyên cho các chương trình

**Câu 2:** **Khái niệm mục tiêu an toàn hệ điều hành**

Mục tiêu an toàn xác định các thao tác có thể được thực hiện bởi hệ thống trong ngăn chặn các truy nhập trái phép.

Các mục tiêu an toàn xác định các yêu cầu mà thiết kế hệ thống cần phải thỏa mã và việc triển khai đúng đắn phải thỏa mã các yếu tố: bí mật; toàn vẹn; sẵn dùng.

+ Bí mật: Hạn chế số lượng chủ thể có thể truy cập vào các đối tượng tài nguyên khác nhau  
+ Toàn vẹn: Đảm bảo nội dung các thông tin tài nguyên cũng như tập các chủ thể có thể truy nhập đến các tài nguyên không bị thay đổi  
+ Sẵn dùng: Hạn chế việc sử dụng tràn lan các tài nguyên của chủ thể và đối tượng tránh làm cạn kiệt tài nguyên

**Câu 3: Khái niệm mô hình tin cậy**

* Mô hình tin cậy của hệ thống định nghĩa tập phần mềm và dữ liệu mà hệ thống dựa vào đó để đảm bảo thực hiện chính xác các mục tiêu an toàn của hệ thống.
* Các phần mềm được tin cậy bao gồm phần mềm xác định mục tiêu an toàn và phần mềm đảm bảo các mục tiêu an toàn này
* Người phát triển HĐH tin cậy phải chứng minh được hệ thống có mô hình tin cậy tồn tại:

+ Phần mềm tin cậy phải thực hiện toàn bộ thao tác nhạy cảm với an toàn

+ Chứng minh tính đúng đắn của phần mềm và dữ liệu tin cậy

+ Chứng minh việc thực thi của phần mềm không bị phá vỡ bởi chương trình khác.

**Câu 4:** **Khái niệm mô hình đe dọạ**

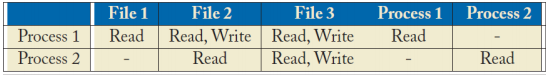
* Mô hình đe dọa xây dựng tập các thao tác mà người tấn công có thể dùng để vô hiệu hóa hệ thống: Tập các thao tác này khong hạn chế theo nghĩa người tấn công có thể áp dụng vào bất cứ thao tác có thể xâm phạm mục tiêu an toàn của hệ thống.
* Nhiệm vụ người xây dựng hệ điều hành an toàn là bảo vệ các phần mềm tin cậy khỏi các dạng đe dọa trong mô hình: chương trình có thể không tin cậy tuy nhiên hệ thống có thể hạn chế việc truy nhập tới dữ liệu nhạy cảm.

**Câu 5:** **Khái niệm hệ thống bảo vệ**

* Hệ thống bảo vệ gồm có:
* Trạng thái bảo vệ mô tả các thao tác mà các chủ thể của hệ thống có thể thực hiện lên các đối tượng hệ thống
* Tập các thao tác trạng thái bảo vệ làm thay đổi các trạng thái này
* Hệ thống bảo vệ xác định các yêu cầu an ninh của HĐH và thực hiện việc quản lý các yêu cầu này qua
* Ma trận truy nhập
* Hệ thống bảo vệ bắt buộc

**Câu 6:** **Ma trận truy nhập**

* Các trạng thái bảo vệ của hệ thóng được biểu diễn bằng ma trận truy nhập, định nghĩa bằng:
* Tập các chủ thể
* Tập các đối tượng
* Các thao tác được phép của chủ thể lên đối tượng
* Ma trận cũng mô tả các thao tác mà chủ thể có thê thực hiệu lên trên ô của ma trận như sở hữu:

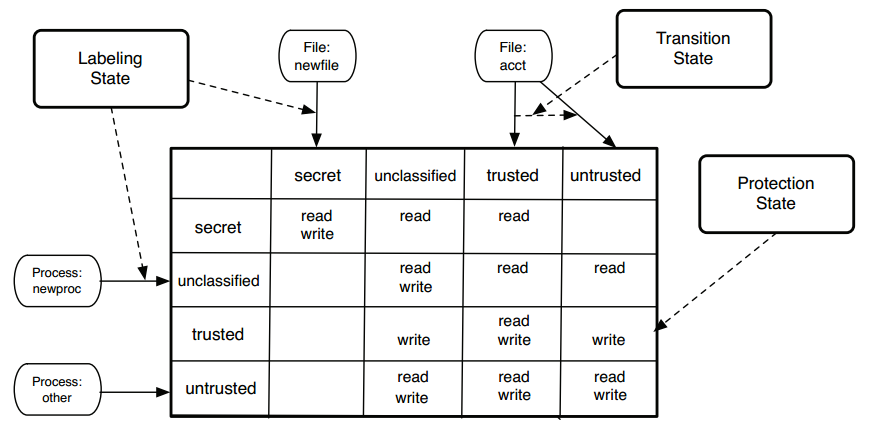


* Ma trận truy nhập cũng được sử dụng để mô tả miền bảo vệ. Trong đó miền bảo vệ là tập các đối tượng (tài nguyên) mà tiến tình có thẻt ruy nhập và các thao tác mà tiến trình có thể dùng để truy nhập tới các đối tượng như vậy.
* Hàng trong ma trận truy nhập cho biết thông tin về miền hoạt động của tiến trình
* Với các hệ điều hành an toàn cần đảm bảo miền an toàn của mỗi tiến trình thỏa mãn các mục tiêu an toàn như tính bí mật hay toàn vẹn.

**Câu 7: Hệ thống bảo vệ bắt buộc**

Là hệ thống mà chỉ có thể được sửa đổi bởi người quản trị tin cậy thông qua phần mềm tin cậy gồm các biểu diễn trạng thái như sau:

* Trạng thái bảo vệ bắt buộc là hệ thống mà chỉ có các chủ thể và các đối tượng được biểu diễn bằng các nhãn. Các trạng thái mô tả các nhãn chủ thể có thể thực hiện lên các nhãn đối tượng.
* Trạng thái dán nhãn để ánh xạ các tiến trình và các đối tượng tài nguyên hệ thống tới các nhãn.
* Trạng thái dịch chuyển mô tả cách thức hợp lệ mà các tiến trình và các đối tượng có thể được dán nhãn lại.



* Trong hệ điều hành an toàn, nhãn chính là các định danh khái quát. Các nhãn này chống phát lại việc xâm nhập (temper-proof) nhờ:
* Tập cá nhãn này được xây dựng bởi người quản trị tin cậy bằng phần mềm tin cậy
* Tập các nhãn không thay đổi được (bởi các tiến trình không tin cậy của người dùng)

**Câu 8:** **Chức năng cơ bản của bộ giám sát tham chiếu (tìm hiểu lại)**

Giám sát tham chiếu là cơ chế thực thi truy nhập cổ điển, khi có yêu cầu truy nhập, bộ phận giám sát trả lời chấp nhận hay từ chối truy nhập, bao gồm: giao tiếp giám sát tham chiếu, module xác thực, kho chính sách:

* Giao tiếp xác định vị trí các truy vấn/ yêu cầu hệ thống bảo vệ được thực hiện tới bộ giám sát. Các thao tác nhạyc ảm về an ninh được xác thực bởi cơ chế thực thi truy nhập.
* Module xác thực là bộ phận cốt lõi, nhận các tham số đầu vào từ giao tiếp như định danh tiến trình, tham chiếu đối tượng, lên lời gọi hệ thống… và thực hiện truy vấn kho chính sách để trả lời tính hợp lệ của truy vấn từ giao tiếp
* Kho chính sách là cơ sở dữ liệu về trạng thái bảo vệ, các nhãn trạng thái và trạng thái dịch chuyển. Các câu truy vấn có cấu trúc {nhãn chủ thể, nhãn đối tượng, tập thao tác} và trả về kết quả nhị phân (hợp lệ/không hợp lệ)

**Câu 9: Khái niệm về nhân an toàn**

Nhân an toàn là cách tiếp cận dựa trên giám sát tham chiếu có kết hợp phần cứng và phần mềm để đảm bảo thực thi các chính sách an toàn của hệ thống.

* Giám sát tham chiếu đảm bảo việc giam sát mỗi truy nhập từ các chủ thể khác nhau của hệ thống tới từng tài nguyên/ đối tượng.
* Cơ sở chính của nhân dựa trên việc vhỉ cso phần nhỏ của hệ thống phần mềm hcịu trách nhiệm về an toàn ngay cả trong hệ thống lớn.
* Các chức năng an toàn được bố trí trong phần lõi (nhân) tin cậy. kích cỡ nhõ của nhân giuaps cho việc kiểm chứng tính đúng đắn của nó được thuật tiện và dễ dàng.
* Phần lõi này phải được bảo vệ chống giả mạo và việc kiểm soát truy nhập của phần lõi này không thể bị bỏ qua.

**Câu 10: Giải thích các thuộc tính của bộ giám sát truy nhập**

* Ngăn chặn hoàn toàn: hệ thống đảm bảo cơ chế thực thi truy nhập ngăn chặn toàn bộ các thao tác nhạy cảm an ninh.
* Chống xâm nhập: không bị sửa đổi các tiến trình không tin cậy
* Thẩm tra được (verifiable): phải đủ nhỏ để có thể kiểm tra và phân tích tính đúng đắn của nó -> phải có khả năng chứng minh hệ thống thực thi các mục tiêu an toàn một cách đúng đắn.

**Câu 11: Nêu và giải thích các nguyên tắc của kiến trúc an toàn**

* Xem xét vấn đề an toàn ngay từ đầu: mức độ quan trọng được đặt ngang bằng như các tính năng vận hành hệ thống. Nếu thiếu quan tâm vấn đề này sẽ không kiểm soát được các phí tổn để bổ sung các tính năng an toàn.
* Lường trước các yêu cầu về an toàn:
* Kiến trúc an toàn cần có tầm nhìn xa đề cập tới các tính năng an toàn tiềm năng thậm chí chưa có kế hoạch sử dụng ngay lập tức -> làm tăng chi phí một chút cho việc nâng cao tính an toàn.
* Lường trước yêu cầu an toàn không chỉ ảnh hưởng đến mức độ cần thiết làm hệ thống an toàn hơn trong tương lai mà còn giúp xác định liệu tính an toàn của hệ thống có thể được nâng cao hay không.
* Vấn đề khác là chính sách an toan. Thay đổi trong chính sách an toàn có thể dẫn tới hậu quả tai hại với các ứng dụng đang hoạt động tốt mà nay xung đột với chính sách mới.
* Giảm thiểu và cách ly các biện pháp an toàn:
* Để đạt được độ tin cậy cao về an toàn hệ thống, người thiết kế cần phải giảm thiểu kích cỡ và độ phức tạp của các phần liên qua tới an toàn của thiết kế. Hệ thống kích cỡ quá lớn làm cho việc nắm bắt tổng thể kém -> hđh ko an toàn.
* Để giảm thiểu các bộ phận liên quan tới an toàn của hđh là chỉ dùng số ít các cơ chế thực thi an toàn. Do đó, bắt buộc các hành động liên quan tới an toàn được giữ trong một số ít phần cách ly. Thực tế điều này khó đạt được điều này, vấn đề an toàn liên quan tới rất nhiều chức năng khác nhau của hệ thống như quản lý hthống, quản lý bộ nhớ.
* Khi cơ chế an toàn đơn giản, dễ nhận biết và cách ly thì dễ dang triển khai các cơ chế bảo vệ bổ sung để tránh các thiệt hại do lỗi tại các phần khác của hệ thống.
* Thực hiện quyền tối thiểu
* Được thể hiện trong các nguyên tắc phát triển hệ thống. Chẳng hạn việc đặt ra tiêu chuẩn lập trình hạn chế các truy nhập tới các dữ liệu toàn cục -> lỗi tác động từ vùng này sang vùng khác.
* Thể hiện trong việc quản trị người dùng và hệ thống. Người dùng và người quản trị không nên được cấp truy nhập nhiều hơn với công việc của họ.
* Giữ các tính năng an ninh thân thiện
* Cơ chế an toàn không được ảnh hưởng tới người dùng, tuân thủ qua định: cơ chế an toàn phải trong suốt với người dùng bình thường. Can thiệp vào công việc hàng ngày làm giảm năng suất và khiến người dùng tìm cách bỏ qua các cơ chế an toàn.
* Thuận tiện cho người dùng để cấp quyền truy nhập. Người dùng cần đc đảm bảo cung cấp đủ truy nhập khi cần thiết và tránh các thủ tục rườm rà và phức tạp
* Thuận tiện cho người dùng để hạn chế truy nhập. Đảm bảo khả năng bảo vệ thông tin người dùng khi cần.
* An toàn không dựa trên bí mật: ngoài việc quản lý mật khẩu, đích chính của kiến trúc an toàn tránh phụ thuộc vào tính bí mật để đảm bảo an toàn.
* Việc giả định người dùng không thể bẻ khóa hệ thống vì ko biết mã nguồn hay tài liệu về hệ thống không an toàn chút nào.
* Công khai mã nguồn hệ thống có thể cải thiện khả năng an toàn nhờ có số lượng người dùng lớn hơn giúp phát hiện và sửa chữa các khuyết điểm.

**Câu 12. Nêu khái niệm về lớp bảo vệ của CPU**

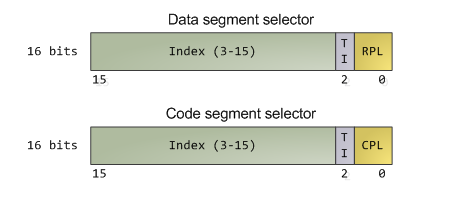
Các lớp đặt ra các ranh giới chặt chẽ và các mô tả những việc mã các chương trình (tiến trình) hoạt động trong từng lớp những thứ được truy nhập và những thao tác được phép thực hiện. Chương trình thuộc lớp trong có nhiều đặc quyền hơn nằm ở lớp ngoài.

* Các lớp tiêu biểu:
* Lớp 0: Nhân hệ điều hành
* Lớp 1: phần còn lại của hệ điều hành
* Lớp 2: Các chương trình điều khiển vào/ra và tiện ích
* Lớp 3: Chương trình ứng dụng

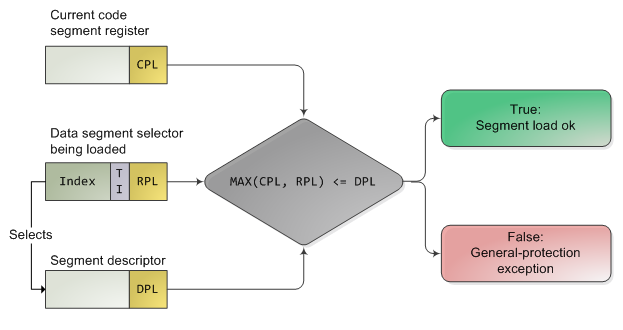
**Câu 13. Giải thích việc thực hiện cơ chế bảo vệ bộ nhớ dữ liệu và lệnh trong tập lệnh x86**

CPU theo dõi mức đặc quyền (lớp bảo vệ) thông qua các trường:

* RPL (Requested Privilege Level – Mức ưu tiên ưu cầu) trên thanh ghi đoạn dữ liệu. Giá trị của trường này không thể được gán trực tiếp bởi các câu lệnh nạp dữ liệu mà chỉ bởi các câu lệnh thay đổi luồng thực hiên chương trình như câu lệnh call



* CPL (Current Privilege Level) trên thanh ghi đoạn lệnh. Giá trị này được duy trì bởi chính CPU và nó luôn bằng với mức bảo vệ hiện thời của CPU. Nói cách khác, giá trị CPL cho biết mức độ bảo vệ của đoạn mã được thực hiện.
* Khi đoạn dữ liệu được nạp việc kiểm tra được diễn ra như trong hình dưới đây: Mức độ bảo vệ của đoạn bộ nhớ DPL (mức đặc quyền mô tả) được so sánh với giá trị lớn hơn giữa CPL và RPL. Giá trị DPL mà lớn hơn thì việc truy cập là hợp lệ



**Câu 14. Nêu khái niệm về toán tin cậy**

* Tính toán tin cậy mô tả các sửa đổi cần thiết về phần cứng và phần mềm để cung cấp nên ftảng ổn định để hệthống máy tính có thể hoạt động trên đó. Hệ thống này có đặc tính
* Độ đảm bảo cao về trang thái (cấu hình, tình hình hoạt động của phần mềm…) của hệ thống máy tính cục bộ. Do vậy có thể xác định khả năng chấp nhận các tác động không mong muốn
* Mức độ đảm bảo cao tương đối về trạng thái của hệ thống ở xa. Thể hiện độ tin cậy của việc tương tác trong hệ thống phân tán.

**Câu 15. Yêu cầu với nền tảng tính toán tin cậy**

* Tính toán tin cậy đòi hỏi thiết kế lại kiến trúc hệ thống sao cho các thành phần riêng lẻ được định nghĩ một cách tường mình các đặc tính của mình: điều này cho phép người thiết kế có thể xác định hành vi của hệ thống.
* Các máy tính được định danh một cách chắc chắn. Sử dụng khóa công khai kèm với khóa bí mật “gắn chặt” liền với hạ tầng tính toán. Cần sử dụng cơ chế phần cứng và chống xâm nhập hay giả mạo.
* Các máy tính xác định chắc chăn cấu hình và định danh chương trình. Sửu dụng mã băm và các cơ chế khác. Phầm mềm, firmware, BIOS, trình nạp, nhân, nhân, chương trình tham gia vào quá trình hoạt động của máy tính cần được kiểm tra thích đang để đảm bảo mức độ tin cậy và thực thi đúng đắn chính sách an ninh mong muốn.

**Câu 16. Trình bày các chức năng cơ bản của mô đun hạ tầng tin cậy TPM**

Mô đun hạ tầng tin cậy được định nghĩa là các chức năng phần mềm (logic) và nhúng vào kiến trúc của máy tính bằng cách sử dụng chíp riêng biệt.

TPM đảm bảo các cơ sở tin cậy sau:

* Cơ sở tin cậy biện pháp bảo vệ (RTM): triển khai một các tin cậy các thuật toán băm chịu trách nhiệm cho các biện pháp bảo vệ đầu tiên với hạ tầng tính toán.
* Cơ sở lưu trữ tin cậy (RTS): triên khai tin cậy vị trí được bảo vệ cho việc lưu trữ một hay nhiều khóa bí mật và một khóa lưu trữ gốc (storage root key – SRK)
* Cơ sở tin cậy cho việc báo cáo (RTR): triển khai tin cậy vị trí được bảo vệ đwr lưu khóa bí mật đại diệnc ho định danh duy nhất của hạ tầng, còn gọi là khóa chứng thực (EK)

**Câu 17. Giải thích cở chế khởi động được bảo vệ (measured boot)**

RTM và RTR là các thành phần căn bản để tạo dựng được sự tin cậy thông qua quá trình khởi động được bảo vệ (measured boot)

Quá trình khởi động như sau:

1. Khi bật máy, RTM lưu lại chỉ số định danh của hệ thống vào vị trí an toàn. Đây có thể chỉ là biện pháp bảo vệ mã của hạ tầng tính toán hay đơn giản chỉ là định danh.
2. Trước khi khởi tạo các phần tử tiếp theo trong chuỗi khởi động RTM tính toán mã băm của bộ phận đó và lưu lại vào nơi an toàn. Sau đó chuyển quyền điều khiển cho bộ phận đó.
3. Lặp lại bước 2 cho từng liên kết trong chuỗi

Như vậy với bất kỳ chương trình nào và bất cứ khi nào đều có thể nhậnđược đảm bro về tính toàn vẹn của bản thân chương trình đó và các chương trì khác tham gia hoạt động của nó.

**Câu 18. Giải thích cách thức lưu trữ an toàn với TPM**

* Lưu trữ an toàn trỏ đến các thanh ghi cấu hình của hạ tầng (Platform Configuratiion Registers – PCR) bên trong TPM
* Các ô nhớ này được bảo vệ bằng cách có thể đọc nhưng ko thể ghi tùy ý.
* Dữ liệu được ghi vào theo dạng tổ hợp với giá trị băm của dữ liệu hiện thời và giá trị trước đó.
* TPM thực hiện việc đóng dấu dữ liệu sử dụng mã hóa công khai với dữ liệu trao đổi.
* TPM cung cấp bản sao có xác nhận trạng thái PCP đảm bảo độ đối tác có thể kiểm tra trạng thái của hạ tầng tính toán. Điều quan trọng là việc xác nhận diễn ra bên trong TPM.

**Câu 19. Giải thích các tác động của tính toán tin cậy>**

* Các tiếp cận của tính toán tin cậy làm thay đổi mạnh mẽ thiết kế của hệ thống máy tính để bàn và ứng dụng phân tán. Tính toán tin cậy đảm bảo chắc chắn phần mềm chạy cục bộ hay ở xa dựa trên cơ chế mã hóa sử dụng cách thức xác thực đảm bảo.
* Tính toán tin cậy ngăn chặn các vụ tấn công dựa trên phần mềm nhờ vào các thao tác thiết yếu cần có sự chứng thực của phần cứng TPM.
* Tính toán tin cậy chịu nhiều chỉ trích không chỉ từ cộng đồng mã nguồn mở
* Tính riêng tư: không bảo vệ định danh người dùng với một số giao dịch
* Kiểm soát của bên bán hàng: bên bán hàng có thể sử dụng TPM khiến cho việc lựa chọn và thay đổi sản phẩm khó khăn hơn với người dùng cuối
* Chứng thực: việc chứng thực sử dụng chữ ký khó khăn do số lượng phần mềm lớn vì vậy việc chứng thực cần thực hiện trên cơ sở hành vi của chương trình.
* Không hỗ trợ mã khóa đối xứng
* Thực thi luật pháp: việc mã khóa mạnh tác động cả hai bên người dùng hợp ệ và người bẻ khóa. TPM mô tả rõ ràng không có cửa hậu trong thiết bị hợp chuẩn.

**Câu 20. Đặc trưng cơ bản của mô hình anh toàn**

Mô hình an toàn có một số thuộc tính cơ bản sau:

* Chính xác và không mơ hồ
* Đơn giản và khái quát do vậy dễ hiểu
* Căn bản: xử lý các thuộc tính an toàn và không hạn chế một cách quá đáng (không thích đáng) va các chức năng hay việc triển khai của hệ thống
* Thể hiện rõ ràng chính sách an toàn

**Câu 21. Giải thích vai trò của mô hình an toàn**

* Mục tiêu phát triển hệ thống nhằm đảm bảo mức độ chắc chắn nhất định rằng việc triển khai hệ thống phù hợp với mô hình lựa chọn
* Mức độ khác biệt về chi tiết giữa mô hình và triển khai thường rất lướn nên cần thêm bước trung gian nhằm đảm bảo sự tương ứng giữa yêu cầu an toàn và triển khai thực tế.
* Các mô tả chính tắc cung cấp các sơ sở cho các chứng minh toán học rằng các mô tả phù hợp với mô hình đề ra.
* Mô hình an toàn có thể dùng như các mô tả về an ninh ch ohệ thống. Việc này giúp hạn chế các lỗ hổng an toàn khi người thiết kế quá tập trung vào chức năng.
* Việc lập mô hình an toàn tiêu tốn nhiều công sức và nhân lực. Các mô hình an toàn đóng vai trò gợi ý cho người thiết kế để phát triển hệ thống phù hợp.

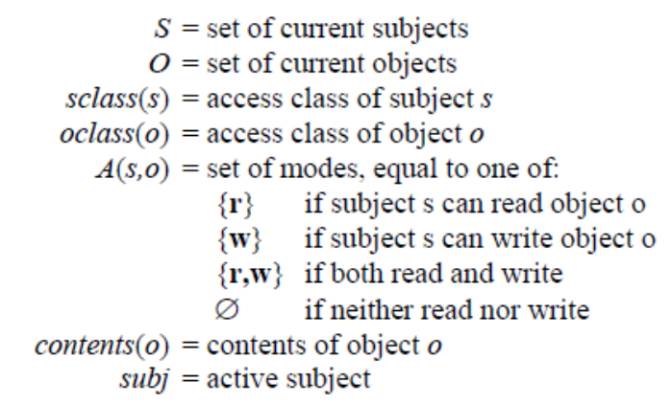
**Câu 22. Trình bày các bước lập mô hình máy trạng thái**

Mô hình máy trạng thái là khái niệm trừu tượng của bit hay byte trong hệ thống thay đổi khi hệ thống hoạt động. Các bước lập mô hình máy trạng thái

1. **Xác định các biến trạng thái liên quan:** Các biến mô tả các chủ thể và đối tượng bên trong hệ thống, các thuộc tính an toàn của chúng cũng như quyền truy cập giữa chủ thể và đối tượng
2. **Xác định trạng thái an toàn:** Mô tả một bất biến biểu diễn quan hệ giữa các giá trị của biến mà luôn được đảm bảo trong khi thay đổi trạng thái.
3. **Xác định hàm chuyển dịch trạng thái**: Các hàm này mô tả các thay đổi tới các biến trạng thái còn gọi là các nguyên tắc hoạt động. Mục tiêu cả các hàm là hạn chế cá thay đổi mà hệ thống có thể thực hiện.
4. **Chứng minh các hàm đảm bảo trạng thái an toàn**: để đảm bảo mô hình nhất quán với cá mô tả ề trạng thái an toàn, cần chứng minh với mỗi hàm hệ thống ở trạng thái ant oàn trước và sau mỗi thao tác.
5. **Xác định trạng thái khởi tạo**, lựa chọn các giá trị cho các biến trạng thái mà hệ thống bắt đầu ở trạng thái an toàn.
6. **Chứng minh trạng thái khởi tạo an toàn theo các mô tả về trạng thái an toàn.**

**Câu 23. Ví dụ máy trạng thái**

* Chính sách: người dùng có thể đọc tài liệu khi và chỉ khi quyền có được lớn hơn hoặc bằng phân loại của tài liệu (classification)
* Mô tả:
* Chủ thể có đọc đối tượng khi và chỉ khi lớp truy nhập của chủ thể lớn hơn hoặc bằng lớp truy nhập của đối tượng.
* Chủ thể có ghi và đối tượng khi và chỉ khi lớp truy nhập của chủ thể nhỏ hơn hoặc bằng lớp truy nhập của đối tượng.
* Mô tả biến trạng thái



S: tập chủ thể;

O: tập đối tượng;

sclass(s): lớp truy cập chủ thể;

oclass(o): lớp truy cập đối tượng o;

A(s,o) = chế độ, bằng một trong các giá trị:

{r}: nếu chủ thể s có thể đọc đối tượng o

{w}: nếu chủ thể s có thể ghi đối tượng o

{r,w}: nếu chủ thể s có thể đọc, ghi đối tượng o

Rỗng; không đọc cũng ko ghi

Contents(o)= nội dung của o

Subj = chủ thể hiện tại

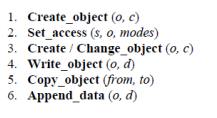
* Trạng thái hệ thống

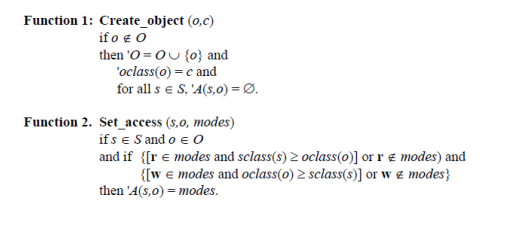


* Trạng thái an toàn

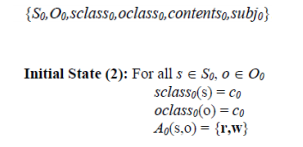


* Các hàm chuyển dịch trạng thái



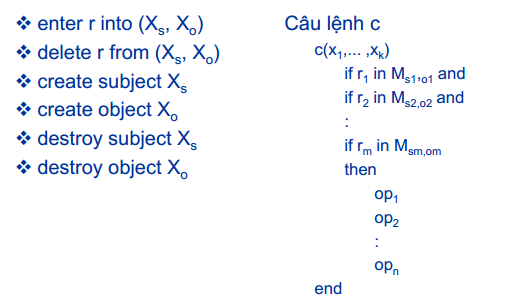


* Trạng thái ban đầu

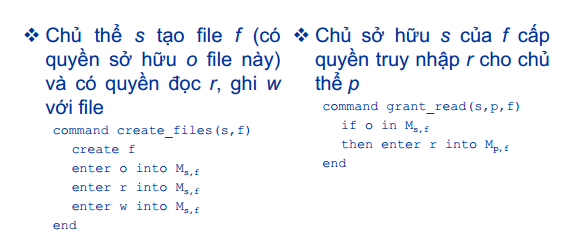


**Câu 24. Trình bày cách lập mô hình HRU? Ví dụ?**

* Mô hình HRU xử lý quyền truy nhập của các chủ thể và tính toàn vẹn của các quyền này. Cho phép quyền truy nhập thay đổi và xác định chủ thể và đối tượng cần được tạo và xóa thế nào.
* Mô hình HRU bao gồm: tập chủ thể S, tập đối tượng O, tập quyền truy nhập R, ma trận truy nhập M (M = Mso)s S, o O | MsoR
* Các thao tác:



Ví dụ:

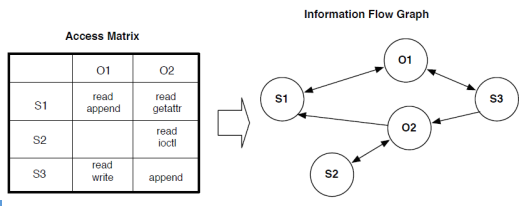


**Câu 25. Các thuộc tính an toàn của mô hình HRU**

* Với ma trận truy nhập M và quyền r, việc kiểm chứng tính an toàn của M với r là không xác định được. Bài toán không giải quyết được trong trường hợp tổng quát đầy đủ. Với mô hình hạn chế hơn, có thể giải thích quyết được.
* Với hệ thống mà các lệnh chỉ chứa 1 thao tác (toán tử) , với ma trận truy nhập M va quyền r, việc kiểm chứng tính an toàn của M là xác định được.. Với hệ thống lệnh chứa 2 thao tác, việc kiểm chứng là không xác định được.
* Bài toán an toàn cho hệ thống xác thực bất kỳ là xác định được nếu số lượng các chủ thể là hữu hạn.

**Câu 26. Cách lập mô hình luồng thông tin? Cách thức xác định các thuộc tính an toàn và toàn vẹn? Cho ví dụ**

* Đồ thị luồng thông tin gồm các đỉnh là các chủ thể và đối tượng, các cung biểu diễn các thao tác giữa cái chủ thể và đối tượng, chiều thể hiện hướng đi của dữ liệu tới đối tượng hay vào bộ nhớ của chủ thể

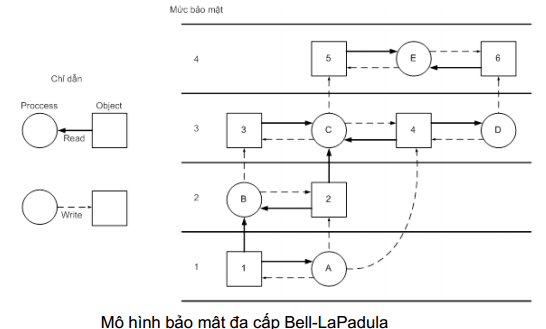


* Cách thức xác định các thuộc tính an toàn và toàn vẹn: Các cung trong đồ thị biểu diễn toàn bộ các đường dẫn mà dữ liệu có thể bị rò rỉ qua đó. Chúng ta có thể dùng đồ thị để xác định liệu có một đối tượng bí mật *o* rò rỉ tới chủ thể khôgn được phép *s*. Nếu tồn tại một đường dẫn từ o tới s thì tính bí mật của hệ thống bị xâm phạm.

**Câu 27. Cách lập mô hình Bell-LaPadula? Đánh giá về thộc tính an toàn của mô hình.**

Quyền truy nhập được định nghĩa thông qua ma trận truy nhập va fthứ tự mức an toàn. Các chính sách an toàn ngăn chặn luồng thông tin đi xuống từ mức an toàn cao xuống mức thấp.

Nguyên tắc của mô hình là không đọc lên, không ghi xuống. Một người dùng ở mức độ bảo mật k chỉ có thể đọc các đối tượng ở cùng mức bảo mật hoặc thấp hơn. Một người ở mức bảo mật k chỉ có thể ghi các đối tượng ở cùng mức hoặc cao hơn.



* Đánh giá:
* Hạn chế cho tính bí mật
* Không có chính sách thay đổi quyền truy nhập
* Chứa kênh ngầm: đối tượng mức thấp có thể phát hiện sự tồn tại của đối tượng mức cao khi bị từ chối truy nhập

**Câu 29. Trình bày đặc trưng và cách xây dựng mô hình Clark-winson**

* Mô hình này tập trưng vào việc ngăn chặn người dùng không hợp lệ sửa đổi trái phép dữ liệu. Trong mô hình này, người dùng không thao tác trực tiếp với các đối tượng mà thông qua một chương trình. Chương trình này hạn chế các thao tác được thực hiện lên đối tượng và như vậy bảo vệ tính toàn vẹn của đối tượng.
* Xây dựng mô hình
* Chủ thể và đối tượng được dán nhãn theo chương trình
* Chương trình đóng vai trò như lướp trung gian giữa chủ thể và đối tượng.
* Việc kiểm soát truy nhập được thực hiện nhờ:
* Định nghĩa các thao tác truy nhập có thể được thực hiện lên từng mục dữ liệu
* Định gnhĩa các thao tác truy nhập có thể được thực hiện bởi chủ thể.
* Các thuộc tính an toàn được mô tả qua cá luật chứng thực và cần kiểm tra để đảm bảo các chính sách an ninh nhất quán với yêu cầu của chương trình.

**Câu 30. Các quy định chứng thực**

* Thủ tục kiểm tra ban đầu IVP phải đảm bảo các mục dữ liệu hạn chế CDI (Constrained Data Items) ở trạng thái hợp lệ khi IVP chạy
* Thủ tục chuyển đổi TP (Tranformation Procedures) phải được chứng thực là hợp lệ tức là CDI bắt buộc phải chuyển đổi thành cDI hợp lệ.
* Các luật truy nhập này phải thỏa mãn bất kỳ yêu cầu về việc tách biệt trách nhiệm
* Tất cả các thủ tục TP phải ghi vào log chỉ ghi thêm
* Bất kỳ TP có dầu vào dữ liệu không hạn chế UDI (unconstrained data items) thì phải chuyển đổi sang dạng CDI hoặc loại bỏ UDI đó và khôg thực hiện việc chuyển đổi nào.

**Câu 31. Quy định thực thi: 4 quy định**

* Hệ thống phải duy trì và bảo vệ danh sách các mục {TP, CDIi, DCIj…} cho phép TP được xác thực truy nhập tới các CDI
* Hệ thống phải duy trì và bảo vệ danh sách {UserID, TPi:CDIi,CDIj,…} chỉ định các TP mà người dùng được chạy
* Hệ thống phải xác thực từng người dugn fkhi yêu cầu thực hiện TP
* Chỉ có chủ thể xác thực quy định truy nhập TP mới có thể sửa đổi mục tương ứng trong danh sách. Chủ thể này phải không có quyền thực thi trong danh sách. Chủ thể này phải không có quyền thực thi trên TP đó.

**Câu 32. Trình bày đặc trưng của đặc tả an toàn**

* Các đặc tả an toàn chỉ hữu ích cho hệ thống cần phải đảm bảo mức độ an toàn/an ninh cao nhất còn mô hình an toàn có khả năng ứng dụng rộng rãi hơn
* Mục đích của đặc tả an toàn là diễn tả các hành vi chức năng của hệ thống theo cách thức chính xác, không lập lờ và phù hợp với việc xử lý của máy tính.
* Các đặc tả chính tắc có thể dùng để chứng minh các thuộc tính về thiết kế của hệ thống đặc biệt là việc phù hợp với các đặc tả của mô hình an toàn.
* Việc kiểm chứng chứng minh việc triển khai tuân thủ hay phù hợp với các đặc tả chính tắc. Việc chứng minh chính tắc đầy đủ cho hệ thống lớn thực sự là thách thức cho dù về mặt lý thuyết chứng minh đã được nghiên cứu rõ ràng.
* Các đặc tả chính tắc trông giống như chương trình máy tính thông thường với biểu thức lô-gíc và tinh toán. Tuy nhiên, các ký hiệu có ngữ nghĩa phong phú hơn ngôn ngữ máy tính cho phép biểu diễn các phép toán lô-gíc và quan hệ
* Các đặc tả chính tắc bao gồm đặc tả giao tiếp và hành vi
* Đặc tả giao tiếp: giúp cho việc phân rã các hệ thống lớn thành các hệ thống con. Các giao tiếp thường được mô tả bằng tập các đối tượng hay thành phần cho biết dữ liệu và các thao tác được truy nhập thông qua giao tiếp
* Đặc tả hành vi: mô tả các trạng thái có thể của hệ thống và các thao tác làm thay đổi trạng thái. Nói cách khác các hành vi của hệ thống có thể được bằng cách xây dựng cách thức các hành vi này làm thay đổi trạng thái của hệ thống như thế nào.

**Câu 33: Trình bày khái niệm máy chứng minh**

Các công cụ chứng minh đặc tả rất phức tạp và có thể có lỗi, các công cụ này là công cụ chứng minh định lý (theorem prover)

* Các hệ thống chứng minh và tích hợp đặc tả sinh ra một cách tự động các định lý dạ trên các tiên đề, hàm, bất biến, các hạn chế và các thành phần khác của đặc tả.
* Cần có sự trợ giúp từ phía người dugf trogn việc sinh ra các tiên đề, ràng buộc hay bất biến.

**Câu 34. Trình bày khái niệm kiểm chứng mô hình (Model checking)**

Kiểm chứng dựa trên mô hình (model based) bằng cách mô tả các hành vi hệ thống có thể theo cách thức chính xác và rõ về mặt toán học. các mô hình hệt thống được kiểm nghiệm tất cả các trạng thái bằng thuật toán.

* Việc này cho phép phát hiện sớm các lỗi như thiếu đầy đủ, mơhồ, không nhất quán trong gia đoạn phân tích thiết kế.
* Kỹ thuật này là cơ sở kỹ thuật từ kiểm nghiệm toàn bộ (kiểm chứng mô hình) hay kiểm nghiệm các tình huống giới hạn (mô phỏng) hay kiểm nghiệm thực tế

**Câu 35. Trình bày khái niệm phân rã cấu trúc dữ liệu?**

Kỹ thuật phân rã (tinh chỉnh) dữ liệu sử dụng nhiều mức trừu tượng với mức độ chi tiết khác nhau

* Mỗi lớp đặc tả là trạng thái mô tả hoàn chỉnh hệ thống
* Lớp trên cùng trừu tượng nhất và kết hợp nhiều kiểu dữ liệu, biến và các hàm vào tron một vài hàm đơn giản
* Các lớp kế tiếp bổ sung các chi tiết bằng các phân rã các hàm khái quát thành các đối tượng và hàm cụ thể.

Kỹ thuật này không cho biết cách thức thiết kế hệ thống bên trong. Việc kiểm chứng cần sử dụng các kỹ thuật côn gnghệ phần mềm truyền thống như kiểm tra mã nguồn và kiểm thử.

**Câu 36. Phân rã thuật toán**

Kỹ thuật phân rã thuật toán cho phép mô tả một phần cấu trúc của hệ thống. Kỹ thuật này coi hệ thống như một chuỗi các máy trạng thái có phân lớp.

* Mỗi máy trạng thái sử dụng các chức năng do lớp trên cung cấp
* Việc triển khai các chức năng, bao gồm các chức năng có trong máy trạng thái ở bên dưới
* Lớp thấp nhất cung cấp các chức năng nguyên thủy nhất cả hệ thống mà không thể phân rã thêm được nữa

**Câu 37. Trình bày sự cần thiết áp dụng kiểm chứng mã chương trình**

Công việc đặc tả cung cấp thông tin ở hai mức

- Mức trừu tượng gần giống với việc lập mô hình

- Mức chi tiết mô tả các thao tác hay hoạt động của hệ thống như mô tả các giao tiếp

Việc chứng minh việc triển hai phù hợp mô hình đề ra có thể vô cùng khó khăn tuy nhiên việc chứng minh mã chương trình phù hợp với đặc tả có thể được chấp nhận như việc chứng minh một phần.

**Câu 38. Khái niệm phân tích tĩnh**

Phân tích tĩnh đề cập đến các kỹ thuật đánh giá mã nguồn nhằm cảnh báo các lỗ hổng an toàn tiềm tàng mà không thực thi chúng.

* Một cách lý tưởng các công cụ tự động có thể tìm kiếm các lỗi an ninh với mức độ đảm bảo nhất định về các lỗi này song việc này vượt quá khả năng của rất nhiều công cụ hiện thời.
* Các kỹ thuật kiểm tra phần mềm (testing) thông thường nhằm kiểm tra các hành vi (chức năng) với người dùng thông thường trong điều kiện thông thường nên rất khó để phát hiện ra các lỗi liên quan đến vấn đề an ninh và an toàn.

Việc đầu tiên công cụ phân tích tĩnh cần làm là chuyển mã chương trình thành mô hình chương trình (program model)

Mô hình chương trình thực chất là cấu trúc dữ liệu biểu diễn đoạn mã.

 Các kỹ thuật phân tích bao gồm

 Phân tích từ vựng (lexical analysis)

 Phân tích câu (parsing)

 Cú pháp khái quát (astract syntax)

 Phân tích ngữ nghĩa (semantic analysis)

 Phân tích luồng điều khiển (control flow)

 Phân tích luồng dữ liệu (data flow)

 Phân tích lan truyền lỗi (Taint propagation)

 Phân tích con trỏ ảo(pointer aliasing)

**Câu 39. Khái nhiệm phân tích động**

* Phân tích động phần mềm được thực hiện bằng cách chạy các đoạn mã trên bộ xử lý vật lý hay ảo: Các đoạn mã ngày càng sử dụng thư viện liên kết động hay theo yêu cầu, việc phân tích tĩnh không thể hiện đầy đủ các khía cạnh của đoạn mã
* Phân tích động nên được thực hiện sau khi hoàn thành việc phân tích tĩnh do việc phân tích động có thể làm tổn hại đến hệ thống
* Để phân tích và đánh giá tính an toàn của hệ thống, có thể cần thực hiện thêm:
* Quét lỗ hổng
* Kiểm thử xâm nhập