HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÀI GIẢNG MÔN HỌC

AN TOÀN HỆ ĐIỀU HÀNH

CHƯƠNG 2 – CÁC CƠ CHẾ AN TOÀN PHẦN CỨNG

Giảng viên: TS. Hoàng Xuân Dậu

Điện thoại/E-mail: dauhx@ptit.edu.vn

Bộ môn: An toàn thông tin - Khoa CNTT1

NỘI DUNG CHƯƠNG 2

1. Đặt vấn đề
2. Hỗ trợ các tiến trình
3. Bảo vệ bộ nhớ
4. Kiểm soát thao tác vào ra
5. Ảo hoá

2.1 Đặt vấn đề

- Để hệ điều hành hoạt động ổn định và hiệu quả, nó cần phải phân biệt được:
  - Các hoạt động của bản than HĐH;
  - Các hoạt động của chương trình người dùng.
- Việc phân biệt các hoạt động khá phức tạp do bản thân HĐH cũng là một chương trình;
- HĐH sử dụng một số cơ chế bảo vệ để kiểm soát toàn bộ các hoạt động và đảm bảo các hoạt động này không vi phạm chính sách an toàn:
  - Một số cơ chế bảo vệ được thực hiện bằng phần mềm;
  - Một số cơ chế bảo vệ được thực hiện bằng phần cứng.

2.1 Đặt vấn đề

- Các cơ chế bảo vệ được thực hiện bằng phần cứng có các ưu điểm so với cơ chế bảo vệ được thực hiện bằng phần mềm:
  - Ổn định và ít lỗi hơn do không bị tác động bởi các phần mềm khác;
  - Thực thi nhanh hơn, cho hiệu năng cao hơn;
  - Các tính năng an toàn tích hợp vào phần cứng giúp kiến trúc hệ thống trong sáng hơn.

2.2 Hỗ trợ các tiến trình

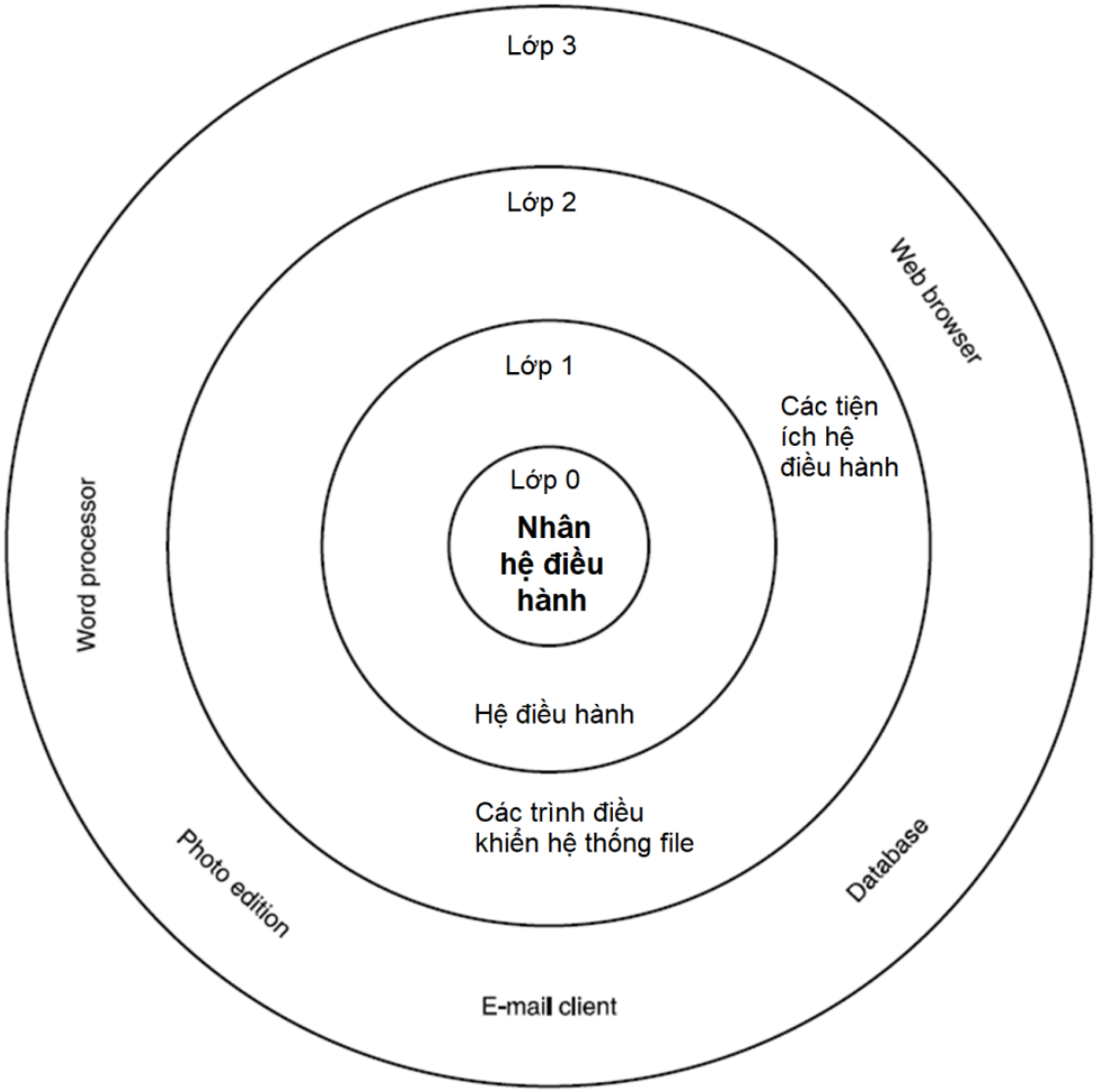
- Yêu cầu cơ bản của HĐH an toàn gồm:
  - Cách ly các tiến trình người dùng với nhau;
  - Hỗ trợ trao đổi thông tin giữa các tiến trình người dùng qua các kênh được kiểm soát.
- Với các HĐH đa nhiệm hiện nay:
  - Các chương trình người dùng chạy song song với nhau nên việc chuyển đổi ngữ cảnh của các chương trình cần được thực hiện một cách nhanh chóng và hiệu quả.

==> Phần cứng (CPU – bộ VXL) giúp làm đơn giản hóa việc chuyển đổi ngữ cảnh thông qua cơ chế "khởi động" lại cách chương trình bị dừng bằng cách lưu lại trạng thái của các tiến trình và khôi phục đồng thời các thanh ghi mà chương trình sử dụng.

2.2 Hỗ trợ các tiến trình

- Các bộ xử lý hỗ trợ việc cung cấp các không gian tách biệt để chạy các chương trình với các đặc quyền khác nhau:
  - Các không gian này thường được biểu diễn như là chế độ hệ thống (đặc quyền) và người dùng (thông thường);
  - Không gian hệ thống có được truy nhập không giới hạn tới các tài nguyên của hệ thống máy tính như toàn bộ không gian nhớ, các câu lệnh;
  - Không gian người dùng bị hạn chế truy nhập tới bộ nhớ và tập hạn chế các câu lệnh.
- Một cách tổng quát, hệ thống phân cấp các không gian thực thi thành các lớp bảo vệ (protection rings).

Lớp bảo vệ không gian thực thi – Mô hình



Lớp bảo vệ không gian thực thi – Mô hình

- Các lớp tiêu biểu trong không gian thực thi:
  - Lớp 0 (Ring 0): Nhân hệ điều hành
  - Lớp 1 (Ring 1): Phần còn lại của hệ điều hành
  - Lớp 2 (Ring 2): Các trình điều khiển vào/ra và tiện ích
  - Lớp 3 (Ring 3): chương trình ứng dụng

Lớp bảo vệ không gian thực thi

- Các lớp bảo vệ đặt ra các ranh giới chặt chẽ và các mô tả các tài nguyên được phép truy cập và các thao tác được phép thực hiện cho mỗi tiến trình hoạt động trong từng lớp.
- Các chương trình nằm ở các lớp bên trong có nhiều đặc quyền hơn là nằm ở lớp ngoài.
  - Lớp có số thứ tự thấp hơn sẽ có nhiều đặc quyền hơn lớp có số lớn hơn.
  - Số lượng các lớp tùy thuộc vào mục đích và nhu cầu cụ thể của hệ điều hành.

Lớp bảo vệ không gian thực thi

- Các thành phần của hệ điều hành hoạt động tại lớp (lớp 0 / lớp 1):
  - Cung cấp truy cập tới vị trí bộ nhớ, thiết bị ngoại vi, trình điều khiển hệ thống;
  - Thực hiện chỉnh sửa các tham số cấu hình hệ thống.
- Chính vì trực tiếp truy cập/sử dụng các tài nguyên quan trọng nên đây là các lớp được bảo vệ chặt chẽ nhất.

Lớp bảo vệ không gian thực thi

- Các tiến trình người dùng chịu sự giám sát và hạn chế truy cập đến bộ nhớ và các thiết bị phần cứng:
  - Tiến trình người dùng gửi yêu cầu truy cập tài nguyên thông qua các chức năng của hệ điều hành hay lời gọi hệ thống;
  - Nếu tiến trình người dùng cố yêu cầu CPU thực hiện các lệnh vượt quá quyền hạn thì CPU xử lý những yêu cầu này như là lỗi hoặc cố gắng khóa tiến trình lại.

Lớp bảo vệ không gian thực thi – Triển khai

- Các lớp bảo vệ được triển khai bằng cách kế hợp giữa phần cứng và hệ điều hành trên thực tế:
  - Phần cứng (CPU/VXL) được cấu hình để hoạt động với một số lớp nhất định;
  - Hệ điều hành được xây dựng sao cho cùng hoạt động ở các lớp này.

Lớp bảo vệ không gian thực thi – Triển khai

- Các lớp bảo vệ hình thành nên:
  - Các rào cản giữa chủ thể và đối tượng;
  - Thực thi việc giám sát truy cập khi các chủ thể thực hiện việc truy cập tới các đối tượng:
    - Mỗi đối tượng và chủ thể được gán một số thể hiện cấp độ của lớp bảo vệ;
    - Chủ thể có cấp độ thấp thì không thể truy cập trực tiếp đối tượng có cấp độ cao hơn.
    - Trong trường hợp cần thiết, chủ thể có cấp độ thấp có thể gửi yêu cầu truy cập đối tượng có cấp độ cao hơn thông qua lời gọi hệ thống. Hệ điều hành sẽ thực hiện việc kiểm soát và hoàn tất thực thi truy cập.

Lớp bảo vệ không gian thực thi

- Việc thay đổi không gian thực thi của chương trình chỉ được thực hiện thông qua lời gọi hàm *call* tới các mục hợp lệ được phép cho trước.
  - Các hệ thống triển khai cơ chế gọi hàm như vậy dưới dạng các bẫy tới các con trỏ hay vị trí xác định trước trong không gian nhớ hệ thống.
  - Với kiến trúc lớp, việc thay đổi lớp (như *R1, R2*) chỉ được hoàn tất thông qua câu lệnh *call* tới vị trí xác định trước trong phần nhớ đặc biệt gọi là *cổng* (gate) mà chúng được gán làm *điểm khởi đầu* cho lớp bên trong.
  - Các cổng được sử dụng để ngăn chặn các tiến trình thực hiện lời gọi hàm vào lớp trong và được thực hiện tại bất kỳ vị trí nhớ nào.

Lớp bảo vệ không gian thực thi

- Cấu trúc mô tả phần nhớ cổng (gate):
  - Mức các lớp được truy cập (R1, R2, R3) phần nhớ
  - Các thao tác được phép thực hiện (R-Read, W-Write, E-Execute)

R1	R2	R3	W	R	E	RING	GATE

Lớp bảo vệ không gian thực thi

- Phương thức truy cập:
  - R1 mức ghi (W): Nếu R1=3 --> các lớp 0, 1, 2, 3 có thể ghi phần nhớ
  - R2 mức đọc (R) : Nếu R2=1 --> các lớp 0, 1 có thể đọc phần nhớ
  - R3 mức thực hiện (E): : Nếu R3=2 --> các lớp 0, 1, 2 có thể thực hiện phần nhớ.
- Số thứ tự lớp hiện thời sẽ chỉ thay đổi khi có phần nhớ cổng và câu lệnh *call* được kích hoạt.

2.3 Bảo vệ bộ nhớ

❖ Yêu cầu bảo vệ bộ nhớ của hệ điều hành:

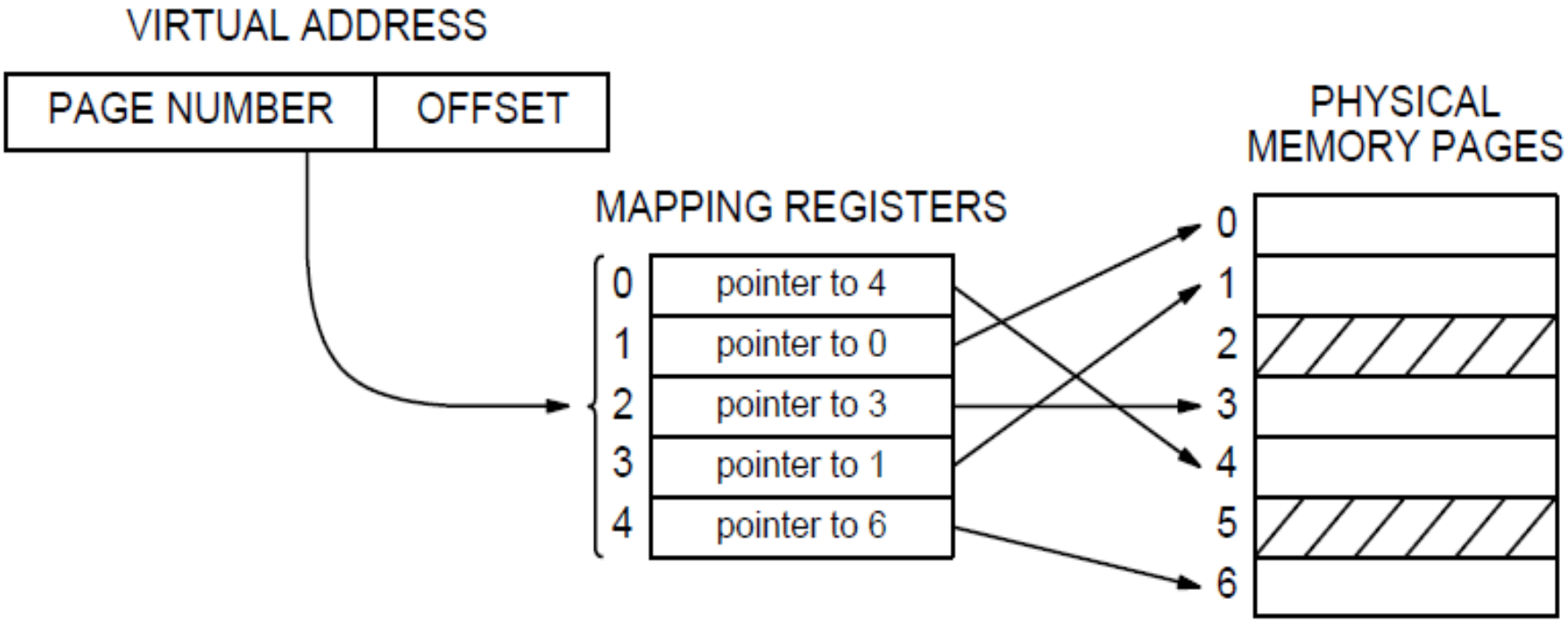
- Các tiến trình người dùng cần được cách ly về không gian bộ nhớ với nhau;
- Các tiến trình người dùng cần được cách ly về không gian bộ nhớ với các tiến trình hệ thống (của bản than HĐH);

==> Hệ điều hành cần phải giám sát và ngăn chặn mọi yêu cầu truy cập trái phép của một tiến trình người dùng đến không gian bộ nhớ của một tiến trình khác.

Bảo vệ bộ nhớ

- Nhiều HĐH hỗ trợ cơ chế ảo hoá bộ nhớ:
  - Không gian bộ nhớ vật ký (bộ nhớ ROM, RAM, đĩa,...) được ảo hoá thành không gian bộ nhớ chung được đánh địa chỉ thống nhất;
  - Việc ảo hoá bộ nhớ trong suốt với tiến trình người dùng:
    - Các tiến trình người dùng truy cập bộ nhớ thông qua bảng chỉ số và con trỏ mô tả phần không gian nhớ lô-gíc của tiến trình;
    - Chỉ có hệ điều hành mới truy cập trực tiếp bộ nhớ nhờ các lệnh sử dụng đặc quyền.

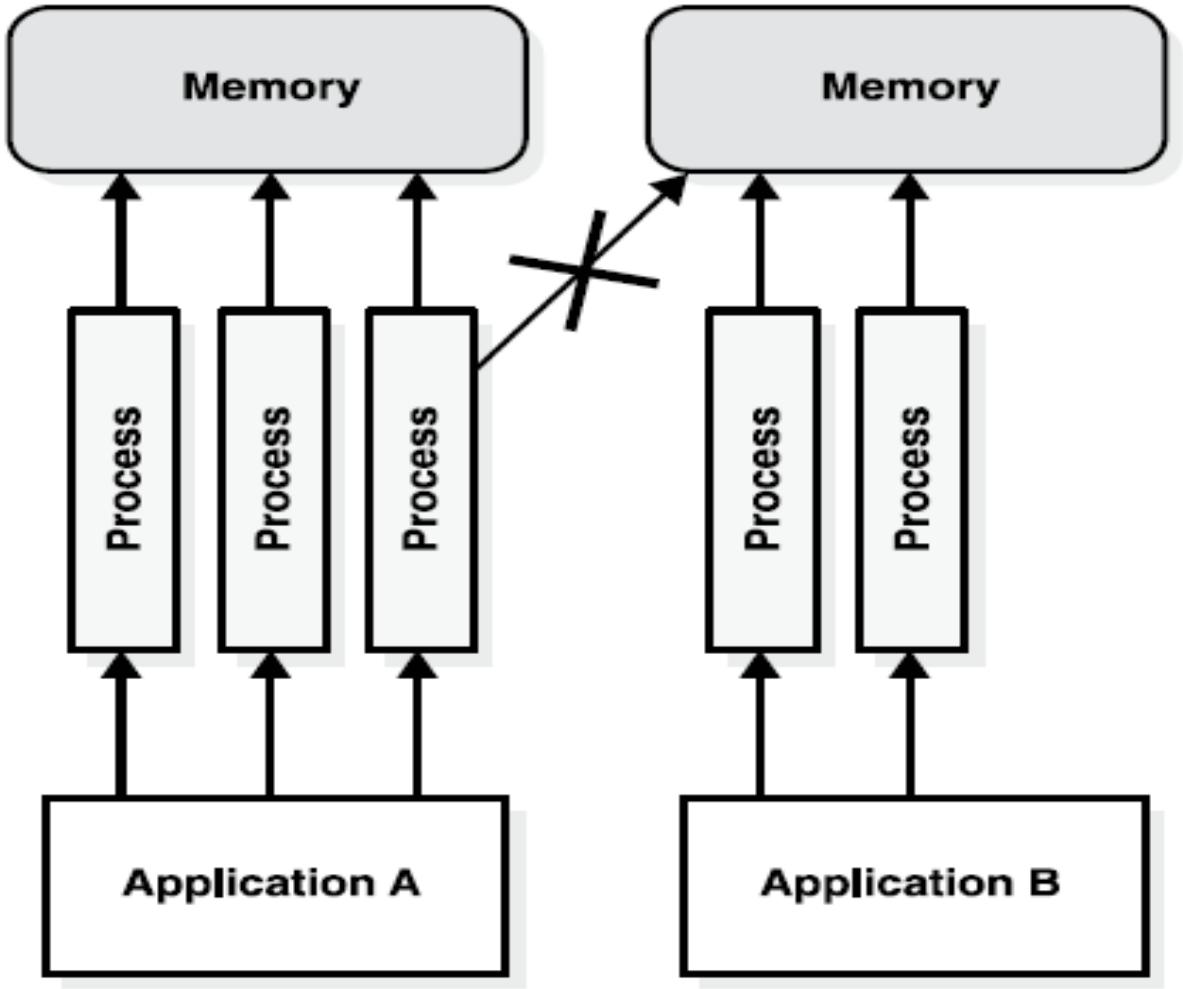
Ánh xạ từ bộ nhớ ảo tới bộ nhớ vật lý theo trang nhớ



Bảo vệ bộ nhớ

- Đặc quyền của các lệnh được bộ xử lý kiểm tra trong quá trình thực thi bằng cách so sánh mức độ đặc quyền của câu lệnh với không gian nhớ được yêu cầu.
- Việc này giúp:
  - Ngăn chặn các truy cập của các tiến trình người dùng nhằm lấy các thông tin về bộ nhớ vật lý của hệ thống;
  - Ngăn chặn các xung đột bộ nhớ giữa các tiến trình người dung.

Xung đột bộ nhớ chương trình người dùng



Các cơ chế quản lý, cách ly bộ nhớ tiến trình

- Trong các HĐH trước đây: không gian nhớ của tiến trình được xác định thông qua con trỏ cơ sở:
  - Cho biết vị trí bắt đầu, và con trỏ giới hạn, xác định vị trí kết thúc;
  - Tiến trình người dùng không thể vượt ra ngoài không gian được cấp.
- Trong các HĐH hiện nay: không gian nhớ của tiến trình được quản lý và cấp phát theo khối nhớ hay trang (page) với kích cỡ hợp lý:
  - Các con trỏ trong các thanh ghi cho biết vị trí bắt đầu của các trang trên bộ nhớ vật lý tùy thuộc theo trạng thái hoạt động của hệ điều hành;
  - Các trang nhớ của tiến trình không nhất thiết phải liên tục.

Các cơ chế quản lý, cách ly bộ nhớ tiến trình

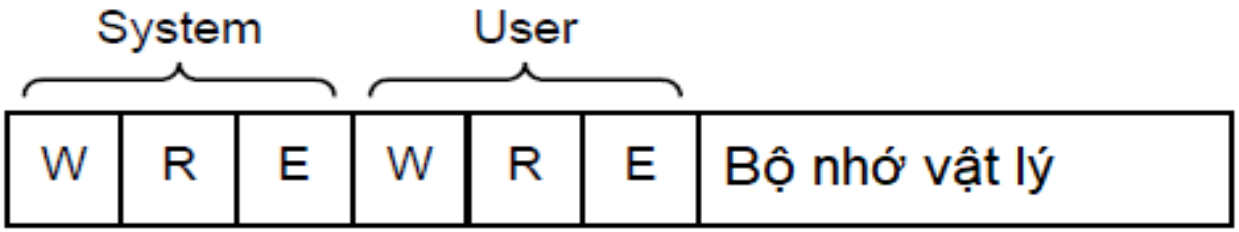
- Khi hệ thống phân cấp các tiến trình theo các lớp bảo vệ (chế độ hệ thống/đặc quyền và chế độ người dùng) thì các tiến trình người dùng không được phép đọc ghi tùy tiện vào không gian nhớ của hệ thống.
- Trong chế độ hệ thống, tiến trình được phép truy cập toàn bộ không gian nhớ vật lý của máy tính.
- Việc chuyển đổi không gian thực hiện của tiến trình được thực hiện nhờ câu lệnh đặc biệt.
  - Để tăng độ linh hoạt, hệ điều hành có thể chỉ định chính xác các phần (trang) của bộ nhớ được phép truy cập tùy theo ngữ cảnh thực hiện tiến trình.

Các cơ chế quản lý, cách ly bộ nhớ tiến trình

- Quyết định cấp quyền truy cập được dựa trên định danh của các trang vật lý:
  - Mỗi trang nhớ được gán nhãn như khóa, các bít truy nhập chỉ định thao tác đọc/ghi;
  - Mỗi tiến trình người dùng khi được gán một khoa;
  - Phần cứng (bộ xử lý) sẽ tiến hành kiểm tra khóa mỗi khi có tham chiếu bộ nhớ: Truy cập được phép chỉ khi khóa truy cập trùng với khóa mô tả đúng thao tác mà người dùng mong muốn.

Các cơ chế quản lý, cách ly bộ nhớ tiến trình

- Với cơ chế chuyển đổi địa chỉ dựa trên các thẻ mô tả (descriptor), mỗi tiến trình có:
  - Tập riêng các thẻ mô tả;
  - Chế độ truy cập của các tiến trình tới trang nhớ được xác định trong các thẻ mô tả: W(Ghi), R(Đọc), E (Thực hiện) – đây là các bít cho biết tiến trình có khả năng truy cập tới vùng nhớ như thế nào.
- Mô tả các chế độ truy nhập bộ nhớ của User và System:



2.4 Kiểm soát thao tác vào ra

- Yêu cầu kiểm soát vào ra
- Các phương pháp phần cứng hỗ trợ kiểm soát vào ra

Yêu cầu kiểm soát vào ra

- Các thao tác vào/ra là các thao tác đặc quyền được thực hiện chỉ bởi hệ điều hành.
- Hệ điều hành thực hiện ảo hoá các thiết bị vào ra, cung cấp các thao tác vào/ra mức cao cho các tiến trình người dung;
  - Tiến trình người dùng không cần thiết kiểm soát các chi tiết của thao tác vào/ra.

Yêu cầu kiểm soát vào ra

- Các thao tác vào/ra liên quan chặt chẽ với việc CPU/Vi xử lý truy cập bộ nhớ;
- Để phần cứng hỗ trợ kiểm soát thao tác vào/ra cần có thêm một số kênh thông tin khác như:
  - Thiết bị vào/ra tới bộ nhớ;
  - Thiết bị vào/ra tới bộ xử lý.
- Việc kiểm soát truy cập tới các thiết bị vào/ra cần phải dựa trên định danh của chủ thể (tiến trình) đại diện cho thiết bị vào/ra được sử dụng và đối tượng (phần bộ nhớ) được sử dụng.

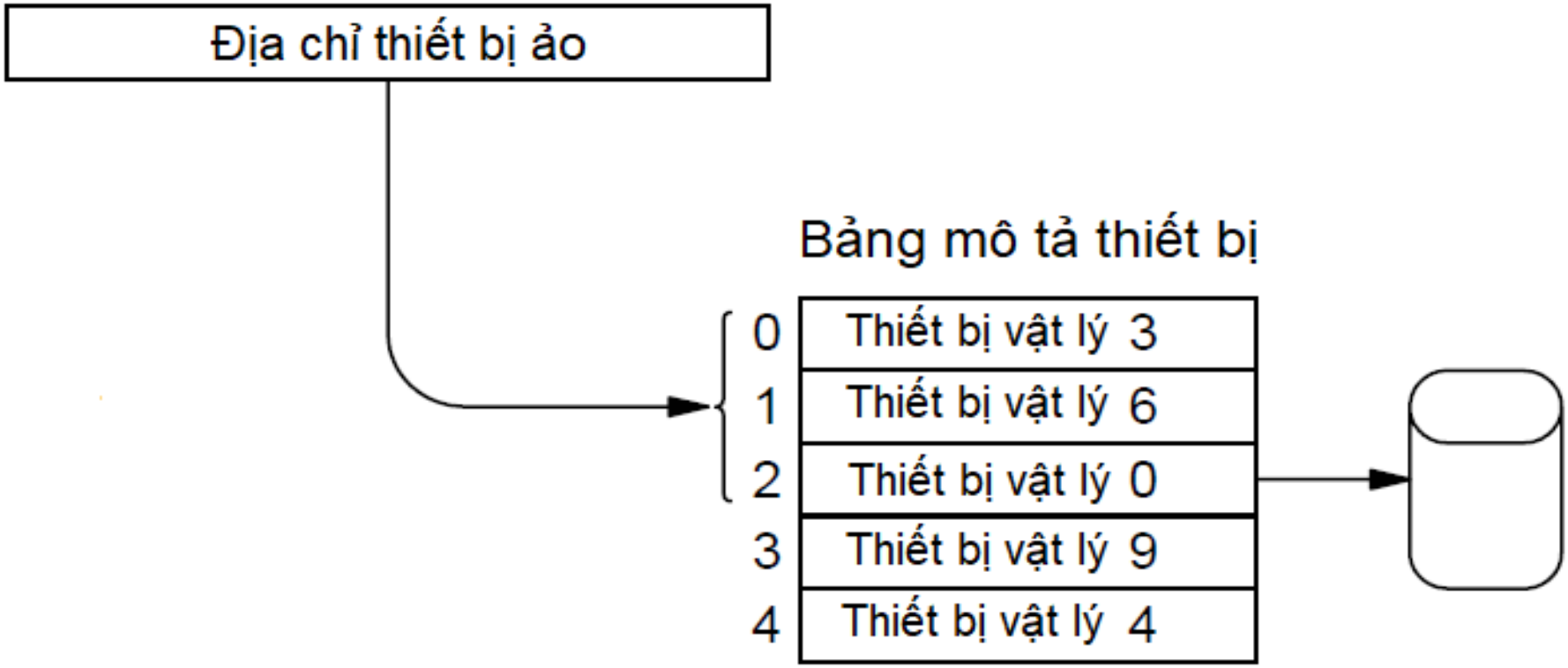
Các phương pháp phần cứng hỗ trợ kiểm soát vào ra

- Lập trình
- Không ánh xạ (unmapped)
- Ánh xạ trước (premapped)
- Ánh xạ đầy đủ.

Hỗ trợ kiểm soát vào ra – Lập trình

- Vào/ra dựa trên lập trình là cơ chế đồng bộ theo nghĩa bộ xử lý kiểm soát trực tiếp từng đơn vị dữ liệu được trao đổi với thiết bị vào/ra;
- Vấn đề an ninh duy nhất là kiểm soát tiến trình yêu cầu vào/ra truy cập tới được thiết bị.
  - Cách trực quan nhất để xử lý vấn đề này là sử dụng bảng mô tả thiết bị (device descriptor table) cho phép ánh xạ tên thiết bị tới thiết bị vật lý cụ thể và kèm theo là các mô tả việc kiểm soát tương tự như kiểm soát bộ nhớ.

Ánh xạ thiết bị ảo tới thiết bị vật lý



Hỗ trợ kiểm soát vào ra – Không ánh xạ

- Vào/ra không ánh xạ thường không cho phép tiến trình người dùng làm việc trực tiếp với các địa chỉ vật lý mà chỉ được kích hoạt từ hệ điều hành.
- Hệ điều hành sẽ chuyển các địa chỉ bộ đệm dùng cho vào/ra từ tiến trình người dùng thành các địa chỉ vật lý.
  - Mặc dù phần cứng hỗ trợ việc chuyển đổi tên thiết bị ảo giống như vào/ra dựa trên lập trình song việc này không giải phóng nhiệm vụ hệ điều hành phải xác thực và thực hiện các thao tác vào/ra.

Hỗ trợ kiểm soát vào ra – Ánh xạ trước

- Vào/ra ánh xạ trước hay ảo cho phép phần mềm xác định địa chỉ bộ đệm ảo.
- Khi các câu lệnh vào/ra được thực hiện, bộ xử lý sẽ:
  - Chuyển các địa chỉ ảo này thành địa chỉ vật lý sử dụng bảng mô tả thiết bị và ánh xạ các thanh ghi của tiến trình hiện thời;
  - Sử dụng địa chỉ vật lý thu được để truy cập thiết bị.

Hỗ trợ kiểm soát vào ra – Ánh xạ trước

- Trong quá trình chuyển đổi địa chỉ, bộ xử lý kiểm tra liệu tiến trình có quyền truy cập hợp lệ tới vị trí đọc ghi.
  - Từ phía thiết bị, các địa chỉ vào/ra là địa chỉ vật lý;
  - Nhưng từ góc độ tiến trình các địa chỉ là ảo và việc kiểm soát truy cập được thực thi nhờ phần cứng.
    - Các thiết bị phải được tin cậy để có thể truy cập tới vị trí mong muốn trong bộ nhớ.

Hỗ trợ kiểm soát vào ra – Ánh xạ trước

- Trường hợp phần cứng không hỗ trợ vào/ra, tiến trình người dùng cũng không thể sinh ra các câu lệnh vào/ra mà không có sự can thiệp của hệ điều hành.
  - Hệ điều hành cần có cơ chế ngăn chặn việc gán lại một cách vô tình (như việc hoán đổi bộ nhớ) các trang nhớ bị ảnh hưởng trong khi vào/ra do tiến trình người dùng khởi xướng đang xảy ra.
  - Các cơ chế ảo hóa giải phóng hệ điều hành khỏi việc thực hiện chuyển đổi địa chỉ và kiểm soát truy cập, nhưng hệ điều hành vẫn phải chịu trách nhiệm quản lý và giám sát các thao tác vào/ra.

Hỗ trợ kiểm soát vào ra – Ánh xạ đầy đủ

- Vào ra với ánh xạ đầy đủ là dạng vào/ra an toàn hơn gồm phần cứng thực hiện việc chuyển địa chỉ từ ảo sang địa chỉ vật lý với mỗi tham chiếu bộ nhớ được thực hiện bởi thiết bị.
  - Thiết bị hoạt động như đối tượng không tin cậy sử dụng các địa chỉ ảo khi đọc ghi thông tin trong bộ nhớ;
  - Phần cứng hỗ trợ việc chuyển địa chỉ nằm trong vùng được bảo vệ thực hiện việc ánh xạ và kiểm tra truy cập;
  - Phần cứng sử dụng cùng các mô tả bộ nhớ thuộc về các tiến trình khởi tạo vào/ra.

Hỗ trợ kiểm soát vào ra – Ánh xạ đầy đủ

- Nhờ việc chuyển địa chỉ và kiểm tra được thực hiện trên từ mức đơn vị dữ liệu nên sẽ không gặp phải vấn đề an ninh khi hệ điều hành phân phối lại bộ nhớ trong quá trình vào/ra.
  - Do việc vào/ra là dị bộ nên địa chỉ ảo của thiết bị vào/ra không nhất thiết phải gắn với không gian nhớ của tiến trình đang chạy.

2.5 Ảo hoá

- Ảo hoá là gì?
- Ví dụ về ảo hoá
- Các vấn đề liên quan đến ảo hoá hỗ trợ kiểm soát truy cập

Ảo hoá là gì?

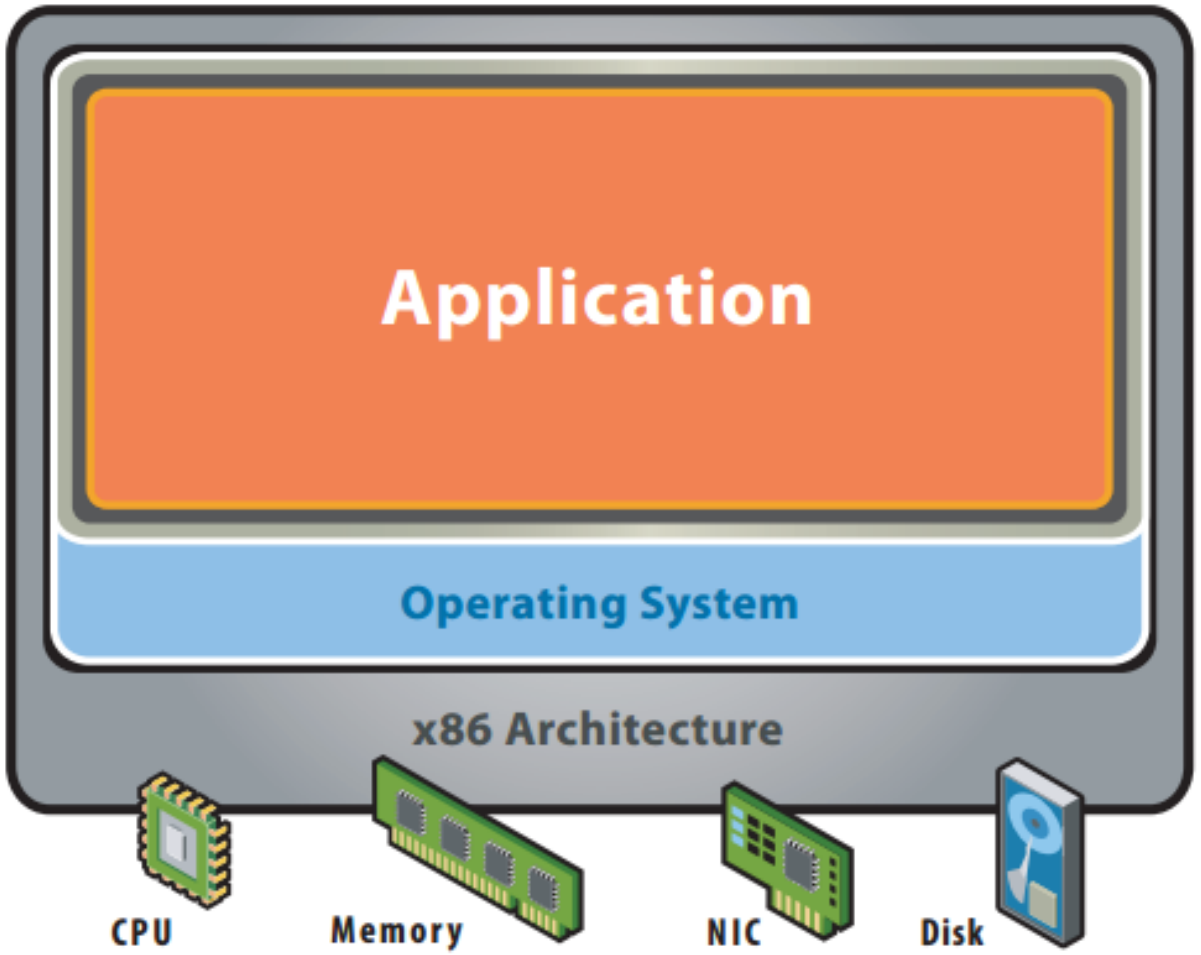
- Ảo hóa (Virtualization) theo nghĩa rộng là sự tách một tài nguyên hoặc một dịch vụ khỏi các phương tiện vật lý dùng để cung cấp nó;
- Ví dụ:
  - Bộ nhớ ảo (Virtual memory): tạo ra không gian bộ nhớ thống nhất cho các ứng dụng từ nhiều loại bộ nhớ vật lý, như RAM, đĩa,…
  - Mạng riêng ảo (Virtual Private Network): 1 mạng riêng được xây dựng trên đường truyền thông công cộng;
  - Máy ảo Java (Java Virtual Machine): một máy thực thi mã Java độc lập với hệ điều hành.

Ảo hoá là gì?

- Ảo hóa (Virtualization) trong tính toán được xem là một hành động tạo ra một phiên bản ảo của cái gì đó, có thể gồm:
  - Phần cứng máy tính;
  - Hệ điều hành;
  - Các ứng dụng;
  - Các thiết bị lưu trữ;
  - Các tài nguyên mạng.

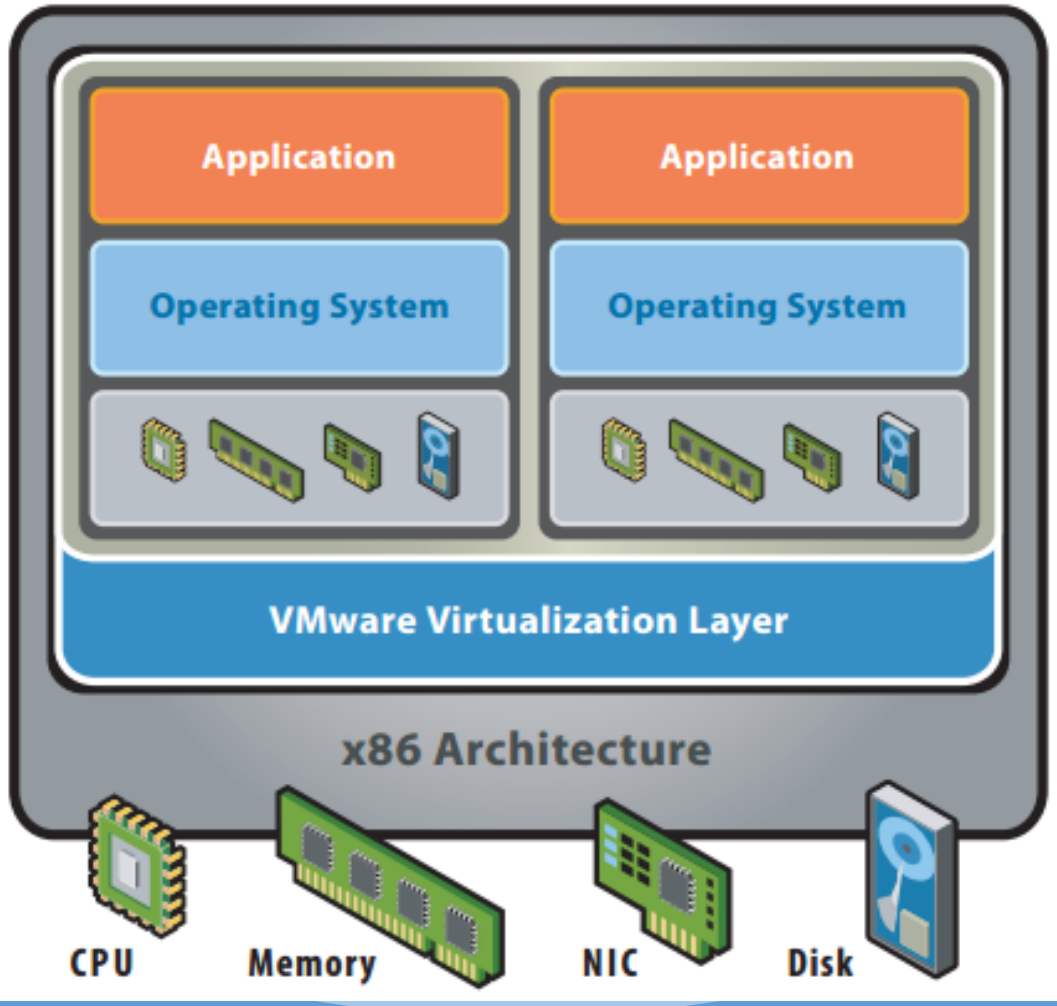
Ví dụ về ảo hóa

❖ Mô hình tính toán truyền thống (không ảo hóa):



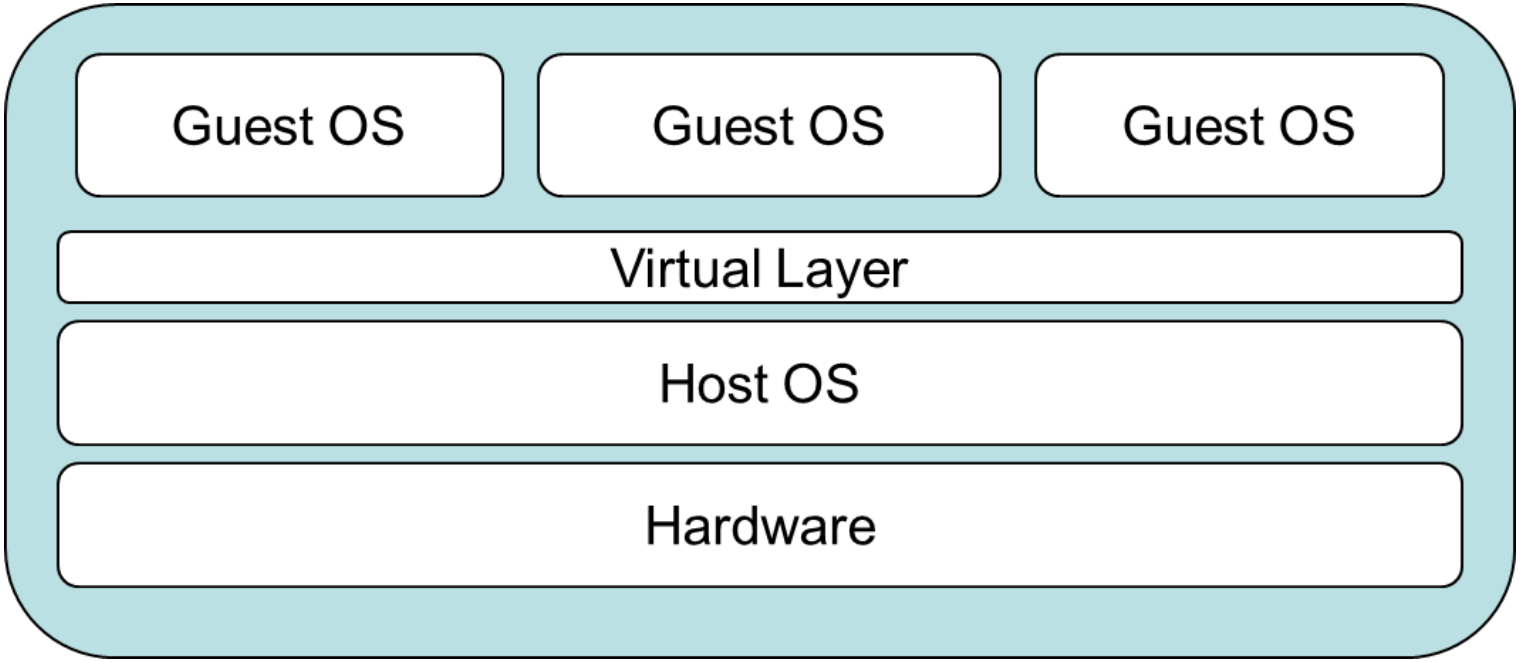
Ví dụ về ảo hóa

❖ Mô hình tính toán dựa trên ảo hóa:



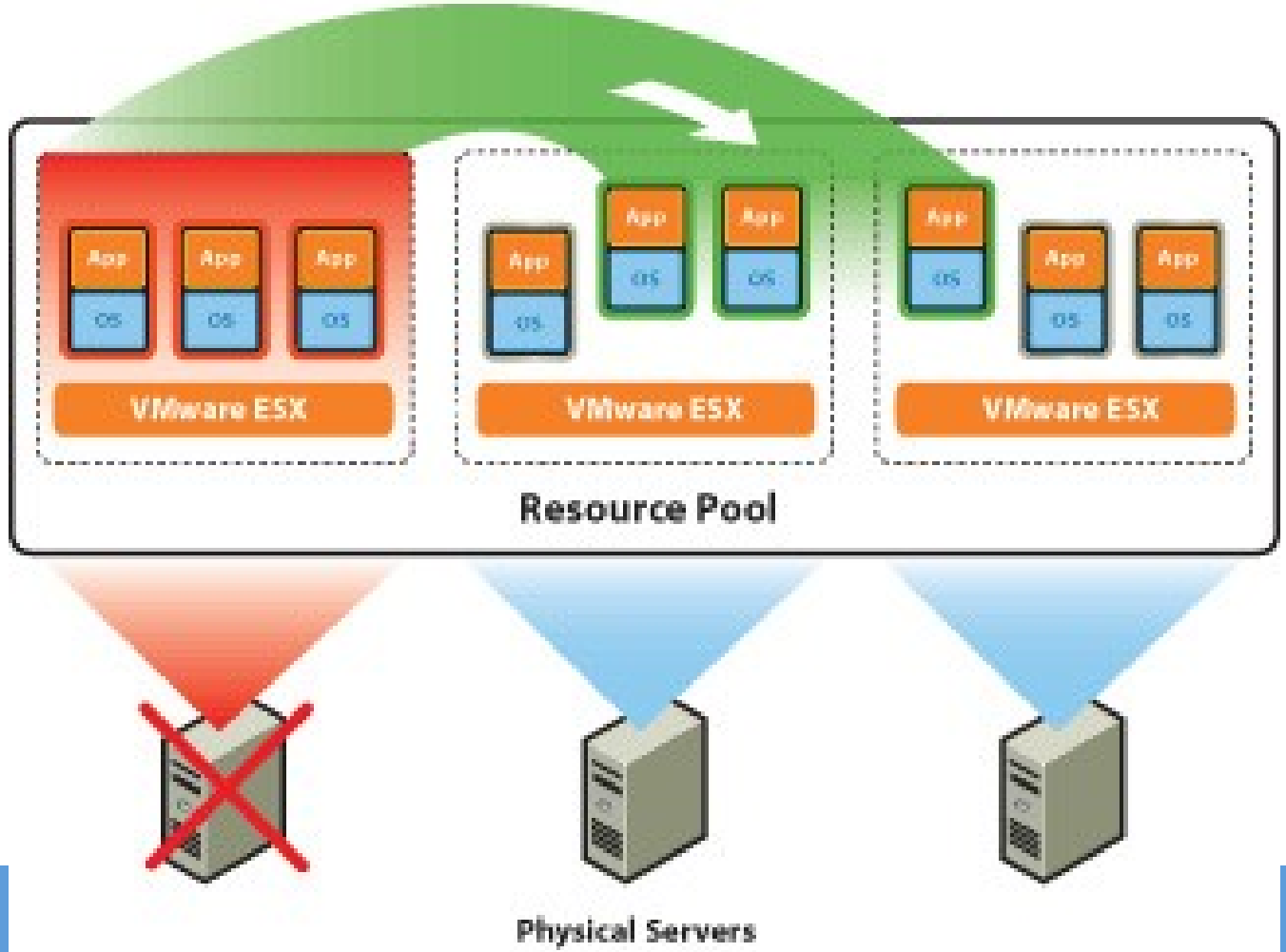
Ví dụ về ảo hóa

- Ảo hóa hệ điều hành (Máy ảo): tách HĐH khỏi phần cứng vật lý:



Ví dụ về ảo hóa

- Chuỗi máy chủ ảo hỗ trợ khả năng chịu lỗi cao



2.5 Ảo hoá

- Ảo hóa giúp cho việc tận dụng và chia sẻ tài nguyên máy tính được thuận tiện và dễ dàng hơn đặc biệt khi năng lực xử lý của hệ thống máy tính được nâng cao:
  - Ảo hoá một số các tài nguyên (CPU, ROM, RAM, đĩa,...)
  - Ảo hoá toàn bộ các tài nguyên (cho phép tạo các máy ảo).
- Phần cứng hỗ trợ ảo hóa giúp cải thiện hiệu năng của các phần mềm ảo hóa và nhờ vậy dễ được người dùng chấp nhận hơn.

Ảo hoá

- Các yêu cầu cơ bản với máy tính cho phép ảo hóa bao gồm:
  - Tính hiệu quả: Tất cả các câu lệnh bình thường được thực hiện trực tiếp bởi phần cứng mà không có sự can thiệp nào của các tiến trình giám sát.
  - Kiểm soát tài nguyên: Không cho phép bất kỳ tiến trình nào ảnh hưởng tới các tài nguyên hệ thống như bộ nhớ và tính sẵn dùng của nó.
  - Bình đẳng: Bất kỳ tiến trình nào đang chạy với sự hiện diện của tiến trình kiểm soát với môi trường thực thi không khác gì trường hợp không có tiến trình giám sát.

Ảo hoá

- Ảo hóa cũng giúp cho việc đảm bảo an toàn cho các tiến trình một cách linh hoạt hơn do các tiến trình người dùng hoạt động trong các không gian cách ly với nhau.
- Tuy nhiên, có vấn đề khó khăn khi thiết kế máy tính có khả năng "bẫy" các câu lệnh có đặc quyền:
  - Các câu lệnh có đặc quyền cao nhất được thực hiện ở lớp 0 và dành riêng cho hệ điều hanh;
  - Các hệ thống ảo chạy trên nền hệ điều hành không thể truy cập đến lớp 0 một cách trực tiếp mà phải thông qua bước chuyển không gian thực hiện (thay đổi đặc quyền).

Ảo hoá

- Ảo hóa phần cứng làm giảm sự can thiệp của hệ thống chủ trong việc xử lý các vấn đề quản lý việc chuyển không gian địa chỉ và đặc quyền.
- Intel với VT-i và AMD với AMD-V là các công nghệ ảo hóa giúp đơn giản hóa hệ thống chủ và đảm bảo hiệu năng gần như thật với hệ thống được ảo hóa.
- Hỗ trợ từ phần cứng cho phép chuyển đổi nhanh chóng giữa hệ thống ảo hóa và bộ phận giám sát (hệ thống chủ) và cấp các thiết bị vào/ra một các an toàn cho các hệ thống ảo hóa.