



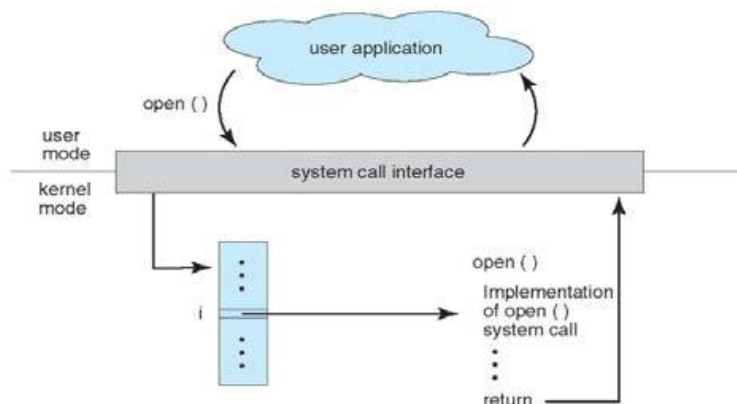
>>> COURSE MATERIAL <<<

INTRODUCTION TO OPERATING SYSTEM / Course ID 502047

Thông tin dưới đây được dịch từ tài liệu StudyGuide tại trang OS-Book của sách giáo trình chính và phần Summary của sách. Mọi lỗi sai sót hay góp ý, xin gửi về cho tôi qua email trantrungtin@tdtu.edu.vn

Study Guide for Lecture notes ch02

- Giao diện người dùng User Interface (UI) có thể là Giao diện dòng lệnh Command-Line (CLI) hay Giao diện đồ họa Graphics User Interface (GUI) hay Màn hình cảm ứng (Touch screen) hay Batch (máy tính giai đoạn trước 1968 khi dùng các bia đục lỗ). Những cách thức đó cho phép người dùng giao tiếp với dịch vụ hệ thống thông qua các lời gọi hệ thống (system calls) (thường được lập trình bằng ngôn ngữ C/C+).
- Các dịch vụ hệ thống (system services) còn lại hữu ích cho người dùng bao gồm: thực thi chương trình (program execution), thao tác nhập xuất (I/O operations), hệ thống tập tin (file-system manipulation), liên lạc (communications) và phát hiện lỗi khi phần mềm chạy (error detection).
- Những dịch vụ cần thiết để đảm bảo tính hiệu quả trong hoạt động của hệ điều hành là: phân phối tài nguyên (resource allocation), kế toán (accounting), bảo vệ (protection) và an ninh (security).
- Phần lớn các lời gọi hệ thống (system calls) có thể được truy cập thông qua các Giao diện lập trình ứng dụng / Application Program Interface (API), ví dụ như là Win32, POSIX, Java.
- Thông thường mỗi lời gọi hệ thống đều có kết hợp với một con số cụ thể.
 - + Giao diện lời gọi hệ thống bảo trì bảng chỉ mục dựa trên các con số đó.
 - + Tham số cần được chuyển cho hệ điều hành, có thể bằng cách: Chuyển giá trị vào thanh ghi, địa chỉ của tham số được lưu trong một khối, đây và stack bởi chương trình và được lấy ra bởi hệ điều hành.



- + Phương pháp Khối [Block] và Ngăn xếp [Stack] không giới hạn số lượng hay độ dài của tham số cần truyền.
- Lỗi gọi hệ thống điều khiển tiến trình (Process control) bao gồm: end, abort, load, execute, create/terminate process, wait, allocate/free memory
- Lỗi gọi hệ thống quản lý tập tin (File management) bao gồm: create/delete file, open/close file, read, write, get/set attributes
- Lỗi gọi hệ thống quản lý thiết bị (Device management) bao gồm: request/release device, read, write, logically attach/detach devices
- Lỗi gọi hệ thống bảo trì thông tin (Information maintenance) bao gồm: get/set time, get/set system data, get/set process/file/device attributes
- Lỗi gọi hệ thống Liên lạc (Communications) bao gồm: create/delete communication connection, send/receive, transfer status information
- Cách tiếp cận phân lớp hệ điều hành (Layered approach) là:
 - + Hệ điều hành được chia là một số lượng các lớp (hay các mức), mỗi lớp được hình thành trên nền tảng của lớp liền kề thấp hơn nó. Lớp nền (mức 0) là phần cứng, lớp cao nhất là giao diện người dùng.
 - + Với tính chất mô đun, các lớp được lựa chọn sao cho mỗi lớp sẽ gọi hàm (hay sử dụng thao tác) và dịch vụ chỉ của các lớp thấp hơn nó.
- Máy ảo (Virtual machine): sử dụng cách tiếp cận phân lớp, uses layered approach, máy ảo xem phần cứng và nhân hệ điều hành như thể tất cả chúng đều là phần cứng.
 - + Chủ (Host) tạo ra ảo tưởng rằng một tiến trình có bộ xử lý riêng và bộ nhớ ảo
 - + Mỗi khách (guest) được cung cấp một bản sao “ảo” của máy tính bên dưới.
- Ứng dụng bị lỗi có thể tạo ra một tập tin lỗi rác (core dump) chụp lại bộ nhớ của tiến trình .
- Hệ điều hành lỗi có thể tạo ra tập tin sập đổ (crash dump) chứa bộ nhớ của nhân.

Summary Chapter 02 of book “OS concepts 10th edition”

- Một hệ điều hành cung cấp môi trường để thực thi chương trình bằng cách cung cấp nhiều dịch vụ đến người dùng và chương trình.
- Ba cách giao tiếp chính để tương tác với một hệ điều hành là (1) trình thông dịch lệnh, (2) giao diện người dùng đồ họa và (3) màn hình cảm ứng.
- Lỗi gọi hệ thống cung cấp một giao diện đến các dịch vụ mà chúng được cung cấp bởi một hệ điều hành. Các lập trình viên sử dụng một lỗi gọi hệ thống trong một API (giao diện lập trình ứng dụng) để truy cập các dịch vụ.
- Các lỗi gọi hệ thống có thể được chia thành sáu loại chính: (1) kiểm soát các tiến trình, (2) hệ thống tập tin, (3) quản lý thiết bị, (4) duy trì thông tin, (5) liên lạc, và (6) bảo vệ.
- Thư viện C tiêu chuẩn cung cấp giao diện các lỗi gọi hệ thống của Hệ thống UNIX hoặc Linux
- Hệ điều hành cũng kèm theo các gói tiện ích hệ thống để cung cấp cho người dùng.
- Một trình liên kết (Linker) kết hợp nhiều mô-đun đối tượng khả tái định vị lại thành tập tin nhị phân khả thực thi duy nhất. Một nạp (Loader) nạp tập tin khả thực thi vào bộ nhớ, khi đó nó sẽ đủ điều kiện để được CPU thực thi.

- Có một số lý do cho thấy các ứng dụng có những đặc trưng riêng của một hệ điều hành. Đó là định dạng nhị phân khác nhau cho chương trình thực thi, các tập lệnh khác nhau trên các CPU khác nhau và các lệnh gọi hệ thống rất khác nhau trên các hệ điều hành khác nhau.
- Một hệ điều hành được thiết kế với các mục tiêu cụ thể. Những mục tiêu này cuối cùng xác định những chính sách của một hệ điều hành. Và hệ điều hành thực hiện các chính sách đó thông qua các cơ chế cụ thể.
- Một hệ điều hành nguyên khối không có cấu trúc; tất cả các chức năng được cung cấp trong một tập tin nhị phân duy nhất và tập tin này chạy trong một không gian địa chỉ đơn nhất. Mặc dù các hệ thống như vậy rất khó sửa đổi, lợi ích chính của chúng là hiệu quả.
- Một hệ điều hành có cấu trúc phân lớp được chia thành một số lớp riêng biệt, trong đó lớp dưới cùng là giao diện phần cứng và lớp cao nhất là giao diện người dùng. Mặc dù các hệ thống phần mềm phân lớp đã có một số thành công, nhưng cách tiếp cận này thường không lý tưởng để thiết kế các hệ điều hành do vấn đề hiệu suất.
- Cách tiếp cận vi nhân để thiết kế hệ điều hành sử dụng một nhân tối giản; hầu hết các dịch vụ chạy như các ứng dụng người dùng. Việc liên lạc giữa các tiến trình đó thông qua các cơ chế truyền thông điệp.
- Phương pháp mô-đun để thiết kế hệ điều hành cung cấp hoạt động dịch vụ hệ thống thông qua các mô-đun có thể được tải thêm hay gỡ bỏ trong thời gian chạy. Nhiều hệ điều hành đương đại được xây dựng lại giữa cấu trúc nguyên khối và cấu trúc mô-đun.
- Một bộ nạp khởi động tải một hệ điều hành vào bộ nhớ, thực hiện các khởi tạo, và bắt đầu thực thi hệ thống.
- Hiệu năng của hệ điều hành có thể được giám sát bằng cách dùng bộ đếm hoặc theo dõi. Bộ đếm là tập hợp các thống kê của các bộ xử lý và cả hệ thống máy tính, trong khi theo dõi sẽ giám sát theo chương trình thực thi thông qua hệ điều hành.