

Lesson 4, 5

Objectives

- Operating system services
- Operating system calls
- System calls categories
- Operating system operations and programs

OPERATING SYSTEM SERVICES

Operating system provides different types of services to users and programs. These services may vary in different operating systems but most common services are listed here. Operating systems provides these services for convenience of users and programmers to make the programming tasks easier. These services are carried out by means of system calls.

1. Program Execution

- *Loading* program into main memory from hard disk
- Fetching the program from main memory to CPU (Fetch)
- Running the program (Execution)
- Ending the program either normally or abnormally (error)

2. I/O operations

- A running program may require I/O either file or device
- A program cannot get directly I/O
- I/O devices may require some operations like settle down the head of HDD
- Operating system provides the means of I/O

3. File-system Manipulation

- Read/write the file
- Creation/deletion of files
- Attributes get/set operations of files

4. Communication

- A process may need to share information with another process
- It may be between processes at same computer or may be between the two at different computers (distributed environment)

- This service can be carried out by:
 - i. Shared memory
 - ii. Message passing (packets)

5. Error detection

- Error detection is a significant service of operation system
- Error may occur at hardware level like in CPU, memory (low memory), power failure
- Error may occur I/O devices like parity error (A parity bit, or check bit is a bit added to the end of a string of binary code that indicates whether the number of bits in the string with the value one is **even or odd**. Parity bits are used as the simplest form of error detecting code.), connection failure at a network, paper out in printer
- Error occurs in user programs
 - i. Arithmetic overflow
 - ii. Attempt to access illegal memory location
 - iii. Excessive use of CPU time
 - iv. Divide by zero error

6. Resource Allocation

- Multiple users have multiple jobs
- Resources are limited so concurrency control is compulsory
- Operating system manages the sharing and allocation e.g., CPU cycles, main memory, file storage etc
- Also called CPU scheduling in order to use best of CPU time

7. Accounting

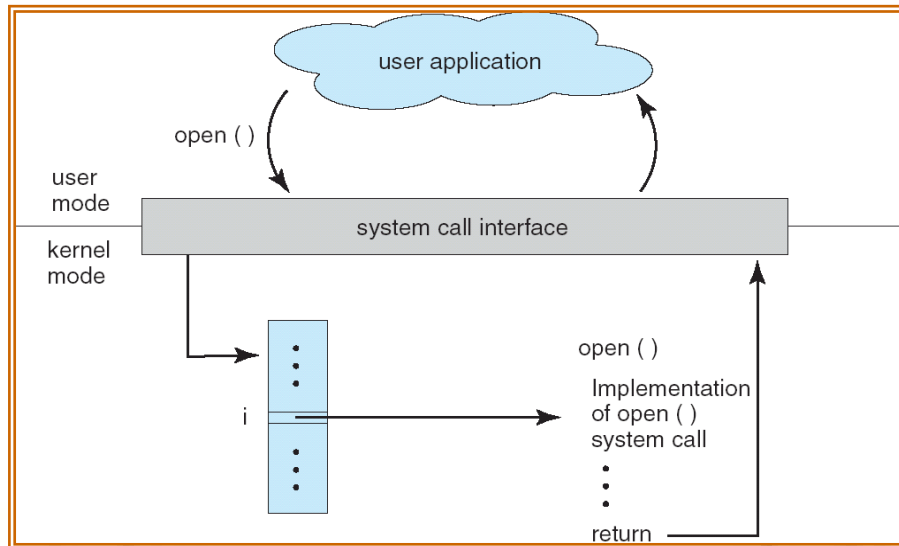
- Keep track which user use how much and what type of resources
- These accounts can be used to bill a user or for statistics
- Also useful for researchers who use to design and configure operating system

8. Protection

- Concurrency control
- Controlled access of resources
- Security from unauthorized use (fire walls)
- Modem and LAN card control from outside intrusion

OPERATING SYSTEM CALLS

- Programming interface to the services provided by the OS
- Typically written in a high-level language (C or C++)
- Mostly accessed by programs via a high-level *Application Program Interface* (API) rather than direct system call use
- Three most common APIs are Win32 API for Windows, POSIX API for POSIX-based systems (including virtually all versions of UNIX, Linux, and Mac OS X), and Java API for the Java virtual machine (JVM)
- Also provides an interface between processes and operating system
For example, we want to read a file in secondary disk location, then system calls involved may be:
 - Read
 - File not found
 - File found but privileged
 - File accessed or opened and close
- A call is sort of parameter sent to operating system
- There are three ways to send parameter:
 - Through registers (DOS)
 - Through a block or table in memory and its address is passed through a register (Linux and Solaris)
 - Parameters can also be pushed onto stack and operating system utilize it by popping out (Windows etc)
- Last two methods are popular and there is no limit of parameters



System calls categories

- *Process control*
 - End/abort
 - Load/Execute
 - Create/terminate process
 - Get/set attributes
 - Wait/signal the event
 - Allocation/free memory
- *File manipulation*
 - Create/delete file
 - Open/close
 - Read/write/reposition/ get & set attributes
- *Device manipulation*
 - Request/release device
 - Read/write/get & set attributes
 - Logically attach/detach the device
- *Information maintenance*
 - Get/set time and date
 - Get/set system data
- *Communication*
 - Create/delete communication connection

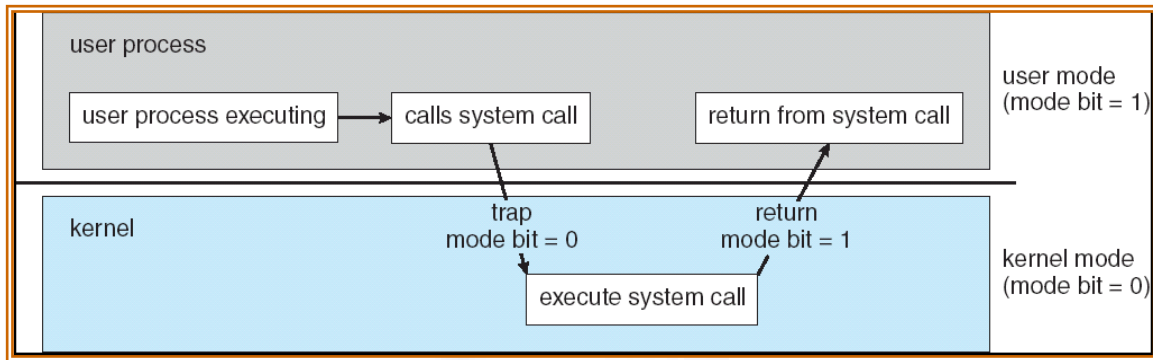
- Send/receive message
- Transfer status information
- Attach/detach remote devices

OPERATING SYSTEM OPERATIONS

1. Interrupt driven by hardware
2. Software error or request creates exception or trap
 - a. Division by zero, request for operating system service
3. Other process problems include infinite loop, processes modifying each other or the operating system
4. Dual-mode operation allows OS to protect itself and other system components
 - a. User mode and kernel mode
 - b. Kernel is a program which always execute when a system turns on
 - c. Mode bit provided by hardware
 - i. Provides ability to distinguish when system is running user code or kernel code
 - ii. Some instructions designated as privileged, only executable in kernel mode
 - iii. System call changes mode to kernel, return from call resets it to user

Transition from user to Kernel mode

- Timer to prevent infinite loop / process hogging resources
 - Set interrupt after specific period
 - Operating system decrements counter
 - When counter zero generate an interrupt
 - Set up before scheduling process to regain control or terminate program that exceeds allotted time



OPERATING SYSTEM PROGRAMS

- Provide a convenient environment for program development and execution
 - Some of them are simply user interfaces to system calls; others are considerably more complex
- File management - Create, delete, copy, rename, print, dump, list, and generally manipulate files and directories
- Status information
 - Some ask the system for info - date, time, amount of available memory, disk space, number of users
 - Others provide detailed performance, logging, and debugging information
 - Typically, these programs format and print the output to the terminal or other output devices.
- File modification
 - Text editors to create and modify files
 - Special commands to search contents of files or perform transformations of the text
- Programming-language support - Compilers, assemblers, debuggers and interpreters sometimes provided
- Program loading and execution- Absolute loaders, relocate-able loaders, linkage editors, and overlay-loaders, debugging systems for higher-level and machine language
- Communications - Provide the mechanism for creating virtual connections among processes, users, and computer systems

- Allow users to send messages to one another's screens, browse web pages, send electronic-mail messages, log in remotely, transfer files from one machine to another

