

* Unit II: Services and Components of OS

2.1) Different operating system services.

- An operating system provides an environment for execution of programs.
- It provides certain services to programs and to users of those programs.

① a) User Interface:

- Almost all operating systems have a user interface, one is commandline interface (CLI) which uses text commands.
- Another is a batch interface in which commands and the directives are entered into files and those files are executed.
- Graphical User interface (GUI) is the window system with pointing device to handle I/O devices, choose from menus and make selection and keyboard is used to enter text.

② b) Program execution:

- The system must be able ^{to} load the program into memory and to run that program.
- The program may complete its execution normally or abnormally (indicating error)

③ c) I/O operation:

- A running program may require I/O to complete its execution (File or I/O device)
- The system must do allocation and deallocation I/O resources

④ d) File System manipulation

- While execution programs need to read, write files and directories.

- They also need to create, delete, rename by their name, search for file, list information of file.
- It performs permission management to allow, deny access to files and directories.

⑤ e) Communication:

- There are many circumstances in which one process needs to exchange information with other process.
- Such communication may occur between processes running on same system or between processes that are executing on different systems.
- This interprocess communication (IPC) is achieved with help of message passing or shared memory mechanism.

⑥ f) Error Detection:

- The operating system need to be always aware of possible errors.
- Error may occur in CPU, memory, hardware, I/O devices.
- For each type of error operating system must take appropriate action to ensure correct and consistent computing.

⑦ g) Resource Allocation:

- When there are multiple users or multiple programs running on system resources must be allocated to each of them.
- Operating system manages many types of resources such as CPU, main memory, Disk, file, I/O etc.

⑧ h) Accounting:

- Keeping the track of which users use how much and what kind of computer resources is known as accounting.
- It is usage statistics, is a valuable tool for researchers who wish to reconfigure the system to improve computing.

⑨ i) Protection and Security:

- Protection involves ensuring that all access to system resources is controlled.
- Security of system from outsiders also important hence each user must authenticate himself/herself to system by means of username and password to gain access.

★ System Calls - Concept:

- System calls provide interface to operating system services.
- Before we discuss how an operating system makes system calls available let's see an example how system calls are used.
- Simple program to read data from one file and copy it to another file.
- First input the program will need is the names of two files (Input file & output file)
- These names can be specified in many ways, one approach is program may ask user for names of two files, this requires sequence of system calls 1) write() a prompting message on the screen and then read() from keyboard.
- On mouse based or icon based system, a menu of file names is usually displayed on window, the user then select source file name and destination file name which requires many I/O system calls.

- Once two file names are obtained, the program must open the input file and create output file. Each of these operation requires system call.
- There are possible error conditions, when program tries to open file, it may find there is no file of that name or file is read protected then program print() message on screen and then terminate() abnormally.
- If input file exist then program must create new output file. We may found that there is already an output file with same name exists, then program must to abort() or we may delete() existing file or ask the user to replace() the existing file
- Now both files are set up, program read() from input file and write() to output file.
- Finally after entire file is copied program may close() both files and terminates() normally
- This way simple program makes heavy use of OS and thousands of system calls are executed in a second.

Invoking open() system call.

- Most systems provides system call interface that serves as link to system calls made available by operating system.
- The system call interface receive function calls through user application (API) and invokes these system calls withing operating system.

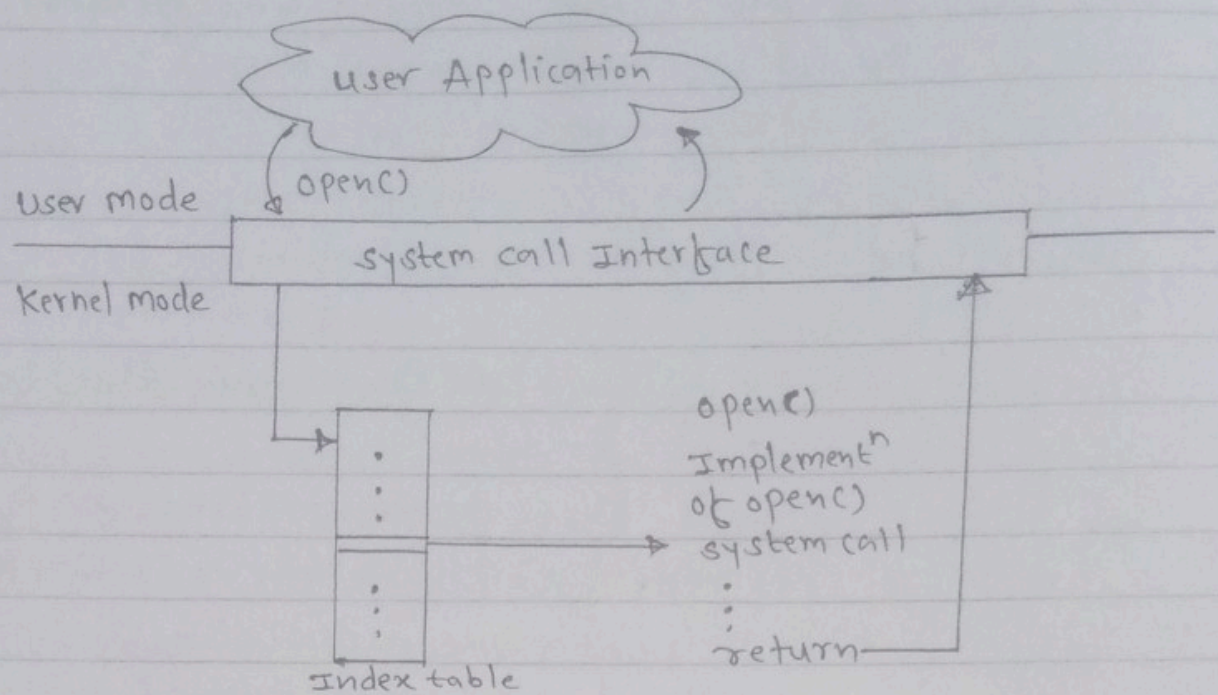


Fig. The handling of user applⁿ invoking openC() system call.

- System call interface maintains a table indexed according to the unique number associated with each system call
- The operating system or kernel invokes the intended system call returns status of system call to user through API
- The caller needs to know nothing about how system calls implemented or what it does during execution.

Types of System calls

- system calls can be grouped into five major categories:
 - 1) process control
 - 2) file manipulation
 - 3) device manipulation
 - 4) Information maintenance
 - 5) communication.

1) Process control

- i) endC(); abortC();
- ii) loadC(); executeC();

- iii) create process C();
terminate process C();

- iv) get process attributes(); set process attributes();
- v) wait for time();
- vi) wait event(); signal event();
- vii) Allocate(); free();

2) File management

- i) create(); delete();
- ii) open(); close();
- iii) read(); write();
- iv) get file attributes(); set file attributes();

3) Device management

- i) request(); release();
- ii) read(); write();
- iii) get device attributes(); set device attributes();
- iv) attach(); detach();

4) Information maintenance

- i) get time or date(); set time or date();
- ii) get system data(); set system data();
- iii) get process, file, device attributes();
- iv) set process, file, device attributes();

5) Communication

- i) create(); delete(); communication connection.
- ii) send(); receive(); messages
- iii) transfer(); status information.
- iv) Attach(); or detach(); remote devices.

* Operating system tools

- Operating system tools makes the user easy & comfortable while using it.

1) User management.

- User management is ability of administrator to define & manage user access to various resources.
- By using user management tool one can create user accounts and define their access policy.
- An administrative user have full access over the all resources
- An administrator can add new user, modify existing user, delete user by using this tool.
- Administrator can create group of users having same kind of access permissions.

2) Device management:

- Using device manager tool we can manage various H/w devices connected to computer system.
- Most of the time when you connect hardware like keyboard, mouse, pendrive, mobile computer detect device and installation of driver done instantly and automatically
- Sometimes devices need to be added manually from add device option in device manager tool
- Once device is shown at the device manager window you can configure, troubleshoot, control.

3) Security Policy:

- This tool allows administrator to provide access or deny access to users to some software or drive.
- Thus changes in those s/w or drive will not be possible to you if you don't have permission.

- An administrator can allow or disallow any user on those computers to get an access over few websites or not
- By this the users can be restricted to surf freely at their workplace as most of irrelevant websites.

4) Performance Monitor

- It is the option to monitor how your system's memory, processor, networking is performing.
- One can go to performance monitor by using control panel or can press combination of keys [Ctrl + Alt + del] and can open task manager.
- Here we can find two graphs, one is showing CPU usage and other showing memory usage
- Through an administrator can monitor performance of a system as all records placed in single window.

5) Task scheduler

- Windows can perform task scheduling and execute that task at scheduled time & date.
- Task scheduler does this by monitoring criteria you choose and executes task when that criteria matches.
- By using task scheduler you can start application, run commands, execute scripts at particular day and time
- You can also trigger a task when a specific event occurs.