

INTRODUCTION TO OPERATING SYSTEMS

SYSTEM CALLS AND SYSTEM PROGRAMS

Unit introduction

- This unit aims at understanding what system calls and interrupts are. It explains basics of operating systems system calls and introduces the aspect of interrupts and interrupts handling.

Learning outcomes:

- By the end of this unit the student should be able to:
- Define what a system call is
- Explain the difference between system calls and system programs
- Explain the concept of interrupts and interrupts handling

System Calls: Introduction

- There are two modes in the computer system
- one is **user mode** and another is **kernel mode**. In computer system there are different types of processes that are running on a computer system. When a user runs an application it is said to be in user mode or computer is in user mode. When there is a requirement of hardware resource, the process sends a request to the kernel to get the process access and then computer enters in kernel mode. These requests are sent by using system call. The computer switches between these two mode frequently. Whenever the task is completed the computer goes back to the user mode from kernel mode. This mode of transition is called **context switching**.

System Calls

- A **system call** is a mechanism that provides the interface between a process and the operating system.
- It is a programmatic method in which a computer program requests a service from the kernel of the OS.
- System call **provides** the services of the operating system to the user programs via Application Program Interface(API).
- It provides an interface between a process and operating system to allow user-level processes to request services of the operating system.
- System calls are the only entry points into the kernel system. All programs needing resources must use system calls.

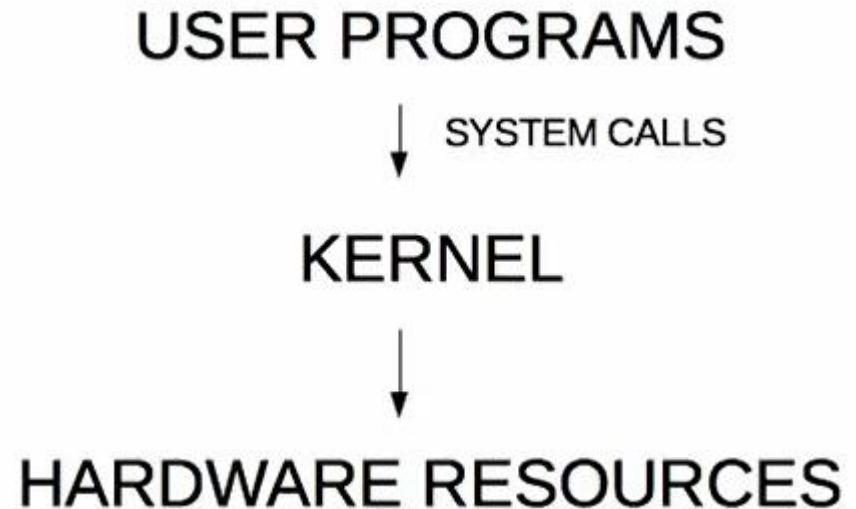
Why do you need System Calls in OS?

We need system calls in OS:

- Reading and writing from files demand system calls.
- If a file system wants to create or delete files, system calls are required.
- System calls are used for the creation and management of new processes.
- Network connections need system calls for sending and receiving packets.
- Access to hardware devices like scanner, printer, need a system call.

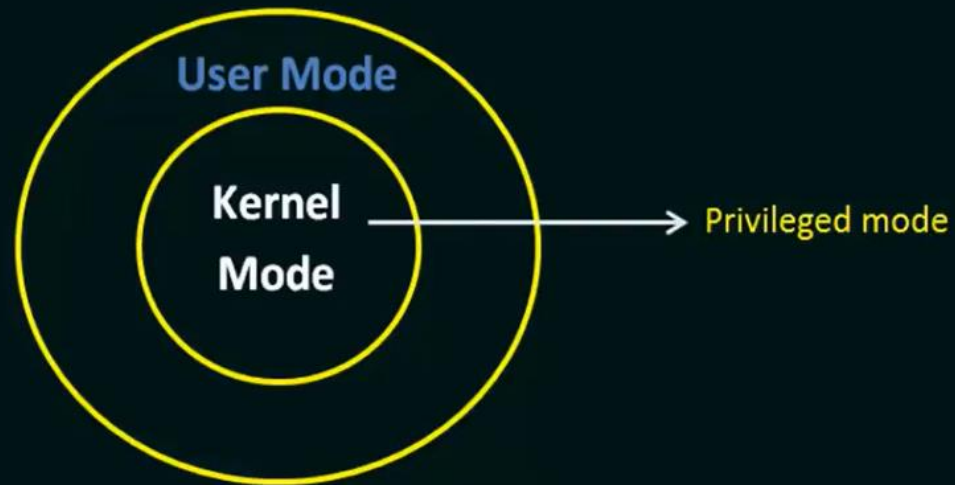
System Calls

- System calls provide an interface between the process and the operating system.
- System calls allow user-level processes to request some services from the operating system, which a process itself is not allowed to do so.
- Interface btn user and OS



System calls

System calls provide an interface to the services made available by an Operating System.



- System call is the programmatic way in which a computer program requests a service from the kernel of the operating system.
- These calls are generally available as routines written in C and C++.



Types of system calls

Here are the five types of System Calls

- Process Control
- File Management
- Device Management
- Information Maintenance
- Communications

Process Control

This system calls perform the task of process creation, process termination, etc.

All processes need to be controlled:

Functions of this call include:

1. Process Control

- end, abort
- load, execute
- create process, terminate process
- get process attributes, set process attributes
- wait for time
- wait event, signal event
- allocate and free memory

File management system calls handle file manipulation jobs like creating a file, reading, and writing, etc.

All systems call relating to file operation and management

Functions:

2. File Manipulation

- create file, delete file
- open, close
- read, write, reposition
- get file attributes, set file attributes

Device management does the job of device manipulation like reading from device buffers, writing into device buffers, etc, including access to i/o devices

- Functions include:

3. Device Manipulation

- request device, release device
- read, write, reposition
- get device attributes, set device attributes
- logically attach or detach devices



These handles information and its transfer between the OS and the user program.

4. Information Maintenance

- get time or date, set time or date
- get system data, set system data
- get process, file, or device attributes
- set process, file, or device attributes



These types of system calls are specially used for inter-process communications, like

Helping OS to transfer status information

5. Communications

- create, delete communication connection
- send, receive messages
- transfer status information
- attach or detach remote devices

Important System Calls Used in OS

(explain in unix wait, fork, exec, kill, exit()s)

| Categories | Windows | Unix |
|-----------------|---|----------------------------|
| Process control | CreateProcess() ExitProcess() WaitForSingleObject() | fork() exit() wait() |

Important System Calls Used in OS

ry:

| es | Windows | Unix |
|--------------|---|---------------------------------|
| control | CreateProcess() ExitProcess() WaitForSingleObject() | fork() exit() wait() |
| manipulation | SetConsoleMode() ReadConsole() WriteConsole() | ioctl() read() write() |
| pulation | CreateFile() ReadFile() WriteFile() CloseHandle() | Open() Read() write() close! |
| ion ence | GetCurrentProcessID() SetTimer() Sleep() | getpid() alarm() sleep() |
| ication | CreatePipe() CreateFileMapping() MapViewOfFile() | Pipe() shm_open() mmap() |
| n | SetFileSecurity() InitializeSecurityDescriptor() SetSecurityDescriptorGroup () | Chmod() Umask() Chown() |