What is system calls and APIs?

1. System Calls:

- A system call is a fundamental mechanism that allows a user-level program to interact with the operating system kernel. It acts as a bridge between user space and kernel space.

- When a program running in user mode needs to perform tasks that require privileged access (such as accessing hardware resources, managing processes, or interacting with files), it makes a system call.

- Examples of system calls include:

  - Creating, executing, or terminating processes.

  - Reading from or writing to files.

  - Managing devices (requesting and releasing them).

  - Retrieving system information (time, date, etc.).

  - Sending and receiving messages.

- System calls are essential for the functioning of an operating system, as they provide a way for user programs to request services from the kernel.

2. APIs (Application Programming Interfaces):

- An API is a set of protocols, routines, and functions that allow communication and data exchange between different applications and devices.

- APIs define how software components should interact with each other. They act as intermediaries, enabling seamless communication.

- Key points about APIs:

  - Purpose: APIs facilitate connectivity among various components, such as web-based systems, databases, operating systems, and software libraries.

- Usage: Developers use APIs to build software by invoking predefined functions and methods.

- Real-World Example: Consider an online travel service that aggregates information from multiple airlines. The travel service interacts with the airline's API to book seats, select meals, and retrieve responses.

- Functionality: APIs take requests from users, inform the system about required actions, and return responses.

- Exchange of Data: APIs allow data exchange between distinct systems, making them essential for modern software development.

3. Difference Between System Calls and APIs:

- System Call:

- A method that allows a program to request services from the operating system's kernel.

- Directly interacts with the kernel.

- Involves context switching between user mode and kernel mode.

- API:

- A higher-level interface that defines protocols, routines, and functions for communication among various components.

- Provides a layer of abstraction above system calls.

- Enables developers to build applications without dealing directly with low-level kernel details.

In summary, while system calls are the low-level requests made by programs to the kernel, APIs provide a more user-friendly way for applications to communicate and collaborate.