What is a Process?

A process is the execution of a program that allows you to perform the appropriate actions specified in a program. It can be defined as an execution unit where a program runs. The OS helps you to create, schedule, and terminates the processes which is used by CPU. The other processes created by the main process are called child process.

What is Thread?

Thread is an execution unit that is part of a process. A process can have multiple threads, all executing at the same time. It is a unit of execution in concurrent programming. A thread is lightweight and can be managed independently by a scheduler. It helps you to improve the application performance using parallelism.

KEY DIFFERENCE

- Process means a program is in execution, whereas thread means a segment of a process.
- A Process is not Lightweight, whereas Threads are Lightweight.
- A Process takes more time to terminate, and the thread takes less time to terminate.
- Process takes more time for creation, whereas Thread takes less time for creation.
- Process likely takes more time for context switching whereas as Threads takes less time for context switching.
- A Process is mostly isolated, whereas Threads share memory.
- Process does not share data, and Threads share data with each other.

User-level threads

All **code** and **data structures** for the library exist in user space.

Invoking a function in the API results in a **local function call** in user **space** and not a system call.

user mode

Kernel-level threads

All **code** and **data structures** for the library exists in **kernel space**.

Invoking a function in the API typically results in a **system call** to the kernel.

kernel mode

Thread Creation / Identification

Just as a process is identified through a process ID, a thread is identified by a thread ID. But interestingly, the similarity between the two ends here.

A process ID is unique across the system where as a thread ID is unique only in context of a single process.

A process ID is an integer value but the thread ID is not necessarily an integer value. It could well be a structure.

A process ID can be printed very easily while a thread ID is not easy to print.

Thread Creation

Normally when a program starts up and becomes a process, it starts with a default thread. So we can say that every process has at least one thread of control.

pthread_create : create a new thread

pthread_t pthread_self(void);

pthread_self : obtain ID of the calling thread

int pthread_create(pthread_t *thread, const pthread_attr_t *attr, void *(*start_routine) (void *), void *arg);

https://www.cs.cmu.edu/afs/cs/academic/class/15492-f07/www/pthreads.html

https://computing.llnl.gov/tutorials/pthreads/

http://www.ccfit.nsu.ru/~fat/pthreads/816-5137.pdf