Process v/s Threads | Operating System – M01 P11

This is a multipart blog article series, and in this series I am going to explain you the concepts of operating system. This article series is divided into multiple modules and this is the first module which consists of 12 articles.

In this article you will the difference between “Process” and “Threads” in an operating system. Here, we are talking about “multiprocessing/multitasking” environment.

Let’s take a general overview about both things. A process is a heavy weight task while a thread is a light weight task.

We will see the difference in between these with the help of a table.

|  |  |
| --- | --- |
| Process | Thread |
| In process we fork the process, and by it a duplicate process starts, which has its own “Data”, “code”, “stack” and “register”  C:\Users\91982\Downloads\Untitled Diagram.png | In thread, if we create multiple threads of a task, then both threads will share the “data”, “code” while they have their own “stack” and “registers”.  C:\Users\91982\Downloads\Untitled Diagram (1).png |
| System calls are involved in process (e.g. fork). Here operating system/kernel is responsible for making multiple processes. | There is no system call involved (generally at user level). Here user or the API is responsible for making multiple threads. |
| Operating system treats different processes differently. | All user level threads are treated as single level for operating system. |
| Different processes have different copies of data, file, and code. | Threads share same copy of code and data. |
| Context switching is slow. | Context switching is faster. |
| Blocking a process will not lock the other process. | Blocking a thread will block entire thread. |
| Independent in nature. | Interdependent in nature. Because they share memory and address. |

So, this was difference between “process” and “thread”. Hope it helped you and you learn something new.

If you have any doubt, question, queries related to this article or just want to share something, then please feel free to contact me.