Lab No-**03**

Name of the lab: **Threads on Operating System**

Name: **Shuvo Biswas** ID: **IT-16014**

***Objective:***

In this lab ,we can learn about Thread , Threads type and Threads

Implementation and how it works in operating system.

**Threads on operating system**

**Q1. What is Thread ?**

**Ans :** **Thread :**

A thread is the unit of execution within a process . A process can have anywhere from just one thread to many threads.

A thread is a basic unit of CPU utilization that shares with other threads belonging to the same process it’s code section, data section and other operating system resources such as open files and signals.

**Q2. Types of Threads :**

Threads are divided into parts. They are given below −

**1. User Threads.**

**2. Kernel Threads**

**User Threads :**

This threads support above the kernel and managed without kernel support. This type of thread library contains code for creating and destroying threads, for passing message and data between threads, for scheduling thread execution and for saving and restoring thread contexts. The main application in this threads starts with a single thread.

**Advantages :**

1. Context switch required no hardware supports.
2. Context switch time is less.
3. Thread switching does not require Kernel mode .
4. User thread can run on any operating system.
5. Implementation of user thread is easy.

**Disadvantages :**

1. In a typical operating system, most system calls are blocking.
2. Multithreaded application cannot take advantage of multiprocessing.

**Kernel Threads :**

Kernel thread is supported and managed directly by operating system. There is no thread management code in the application area. All of the threads within an application are supported within a single process. Kernel threads are generally slower to create and manage than the user threads.

**Advantages :**

1. Kernel can simultaneously schedule multiple threads .
2. If one thread in a process is blocked, the Kernel can schedule another thread of the same process.
3. Kernel routines themselves can be multithreaded.

**Disadvantages :**

1. Hardware supports is needed.
2. Implementation of kernel thread is complicated.

**Q3. How to implement a Thread ?**

**Ans:**

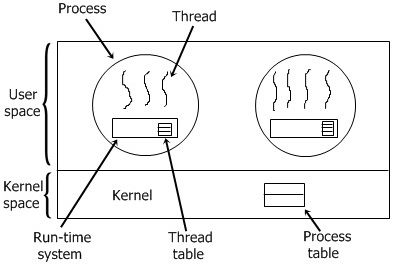
There are two ways of implementing a thread :

1. Threads implementation in user space
2. Threads implementation in kernel

Now describe briefly about the above two ways of implementing a thread .

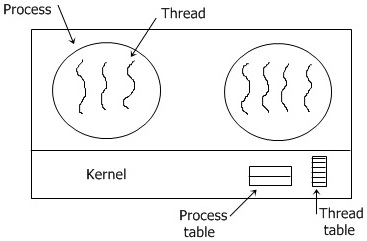
1. **Threads implementation in user space :**

In this model of implementation, the threads package entirely in user space, the kernel has no idea about it. A user thread package can be executed on an operating system that does not support threads and this is the main advantage of this implementation model. All of these implementations have the same general structure as illustrated in the figure given below.



1. **Threads implementation in user space :**

In this method of implementation, the threads package entirely in the kernel, no any run-time system is need in each as illustrated in the figure given below.



In this, there is no any thread table in each process. But to keep track of all the threads in the system, the kernel has the thread table .Whenever a thread wants to create a new thread or destroy an existing thread, then it makes a kernel call, which does the creation or destruction just by updating the kernel thread table .The thread table of the kernel holds each registers, state, and some other useful information of the thread. In this method of implementation model, the threads package completely in the kernel. There is no need for any runtime system. To maintain the record of all threads in the system a kernel has a thread table.

**On the other hand, we can implement thread in two methods. They are given bellow:**

1. Hybrid implementation
2. Scheduler activation

**Hybrid implementation**

In this implementation, there is some set of user-level threads for each kernel level thread that takes turns by using it.

**Scheduler activation**

The main purpose of this scheduler activation is to replicate the function of kernel threads, but with higher performance and better flexibility which are usually related to threads packages which are implemented in userspace.

**Conclusion :**

By doing this lab report, I learnt about the basic concept of thread, thread types, how it can be implemented and also how it works in operating system by using kernel. Mainly, threads are used multiple application running at a same time period in a processor. The main benefit of using thread is that we can do multiple task by dividing a process into multiple threads.