



# **Mawlana Bhashani Science and Technology University**

## **Lab-Report**

Lab Report No: 03  
Lab Report Name: Threads on Operating System  
Course code: ICT-3110  
Course title: Operating System Lab  
Date of Performance:  
Date of Submission: 10/09/2020

### **Submitted by**

Name: Ali Ashadullah Arif  
ID:IT-18031  
3<sup>rd</sup> Year 1<sup>st</sup> Semester  
Session: 2017-2018  
Dept. of ICT  
MBSTU.

### **Submitted To**

Nazrul Islam  
Assistant Professor  
Dept. of ICT  
MBSTU.

## Lab No - 03

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

#### 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?**

##### **Answer:**

**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?**

##### **Answer:**

Threads are divided into parts. They are given below –

1. User Threads.
2. Kernel Threads

**User Threads:** These threads support above the kernel and are 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 these 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 threads is easy.

#### **Disadvantages:**

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

**Kernel Threads:** Kernel-level threads are handled by the operating system directly and the thread management is done by the kernel. The context information for the process as well as the process threads is all managed by the kernel. Because of this, kernel-level threads are slower than user-level 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?**

#### **Answer:**

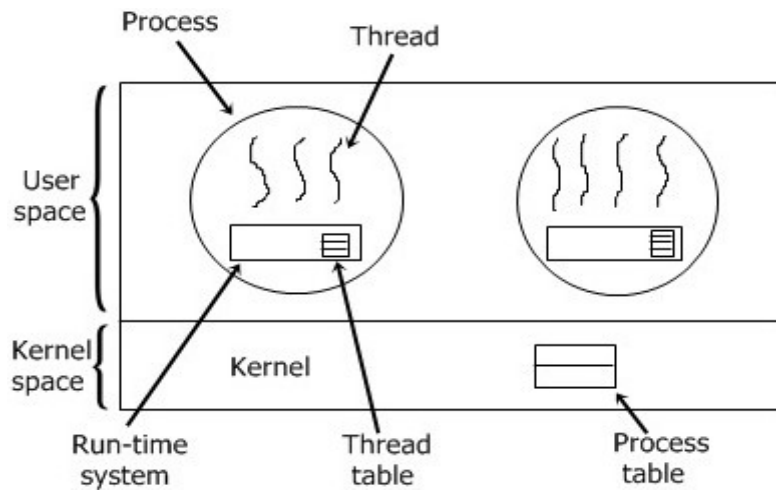
There are two ways of implementing a thread :

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 :**

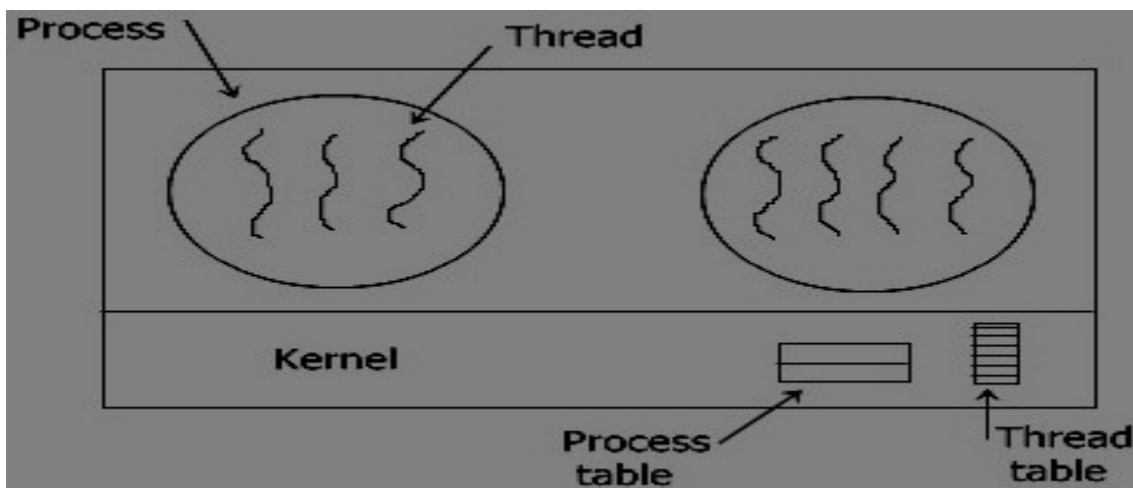
In this model of implementation, the threads package entirely in user space, the kernel has no idea about it. A user thread package can be executed on an operating system that does not support threads and this is the main advantage of this implementation model. All of these implementations have the same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



**Fig: Threads implementation in user space**

## 2. Threads implementation in kernel:

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.



**Fig: Threads implementation in Kernel**

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

run-time system. To maintain the record of all threads in the system a kernel has a thread table.

**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.