Module Code & Module Title

Level 5 – Network Operating System

Assessment Type

Logbook 1

Semester

2024 Spring

Student Name: Yogesh Dhakal

London Met ID: 23057067

College ID: NP04CP4S240032

Assignment Due Date: 30th September 2024

Assignment Submission Date: 30th September 2024

Submitted To: Mr. Prashant Adhikari

Word Count (Where Required):

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Contents

1. Introduction

kernel is the core of an operating system, which provides a system's basic functions to manage its resources and hides complex related details from other parts of the OS. The kernTheel, as it operates between user level applications and the low-level hardware is responsibile for accessing critical hardware components such as the CPUs, memory, and storage. It acts as a referee, making sure programs are talking to hardware appropriately so they can do their job in an efficient and secure way while keeping the system stable. Because it backbones the demands of an operating system, a Kernel is such fundamental factor in general design.

Major components and Functions of a kernals:

i.) **Process Management**: Process-related tasks, such as creation, scheduling, termination, deadlock, etc., are included in process management.

ii.) **Memory Management**: An operating system's ability to handle or manage primary memory and switch between main memory and the disc during the execution of a task is known as memory management.

iii.) **Device Management**: The operating system's device management feature regulates a computer or PC's hardware or virtual devices to ensure optimal operation.

iv.) **File System Management**: The procedures and methods involved in producing, organising, accessing, modifying, and managing files stored on storage media are referred to as file management in an operating system.

v.) **Security**: The operating system is responsible for overseeing memory, the file system, and device security.

vi.) **Networking**: The collection of programs, instruments, and procedures used to establish, run, maintain, oversee, and safeguard network infrastructure is known as network management.

2. Objective

Types of kernel:

Based on Structure:

i.) **Monolithic Kernel**: The most basic and prevalent kind of kernel is a monolithic kernel. They support any device that is linked to the operating system and contain all of its essential features. Here, the memory space used to implement kernel and user services is shared. As a result, the kernel grows larger, increasing the operating system's overall size. The primary advantage is in the quicker process execution caused by the elimination of separate memory spaces for the kernel and user.

ii.) **Microkernel**:

Monolithic kernels are more frequent than microkernels since they are a more recent invention. They just have the necessary components and services that the system needs in order to operate. Smaller, quicker, and using less memory is the outcome of this. This implementation separates the kernel and user services into two areas. User and Kernel Spaces are independent. The operating system's size is subsequently reduced as a result of the kernel's reduction.

iii.) **Hybrid kernel**

Monolithic and microkernels are combined in hybrid kernels. Compared to microkernels, they offer more services, but not as many as monolithic kernels. They are able to provide some advantages of both kernels as a result. It takes modularity from microkernels and speed from monolithic kernels.

iv.) **Nano Kernel**:

Nano kernels, with its few thousand lines of code, are the smallest kind of kernel. This indicates a low amount of code running in the hardware's privileged mode. In embedded systems or low-resource devices, they are usually utilised.

v.) Exo Kernal:

Resource management and protection are independent in this kernel. It can be used for customisation tailored to a given application. Exo kernels are intended for usage in portable electronics. They are a type of microkernel with extra capabilities designed especially for mobile devices, such support for multiple CPUs and power management.

3. Popular Kernals and their History

i.) IOS

This kernel is Mach-based (while it has components from BSD Unix) and was inherited by iOS. XNU is based off of the Mach project at Carnegie Mellon University in the 1980s and later adapted by NeXT for their NeXTSTEP operating system. The XNU kernel was acquired by Apple in 1997 and evolved to work in macOS first and iOS later on tasks demanded for mobile devices.

ii.) **Windows**:

Windows is built out of the NT kernel and it was released in 1993. The (NT) kernel, designed at Microsoft by Dave Cutler, wasn't the first to be developed in Redmond, but it was the first to be created with an eye toward high security and scalability—characteristics relatively new in commercial operating systems—and complete portability. Introduced with Windows NT 3.1 in 1993, it eventually underpinned all versions of the modern Windows operating system from Windows 10 and now through to Windows 11.

iii.) **Ubuntu**:

The Linux kernel on which Ubuntu is based, is a monolithic kernel. In 1991, Linus Torvalds developed the Linux kernel as a free and open-source project with the intention of creating a non-proprietary Unix-like operating system. What started as an under horse project has blossomed into the basis of many Linux distributions, one of which is now the largest — Ubuntu. Linux kernel is modular and prepared for a number of device types, ranging from the Internet servers all the way down to smartphones.

4. Boot Process

i.) iOS (XNU Kernel):

System: Hybrid kernel that is based on parts of the Mach microkernel, and of BSD Unix.

History: XNU was developed by Xerox with Macintosh at Carnegie Mellon with Mach project and then Apple after buying NeXT in 1997.

ii.) Windows (NT Kernel):

Type: Hybrid kernel combining monolithic and microkernel features for performance and security.

• History: Microsoft built it in the 1990s under the leadership of Dave Cutler, and DEC's VMS served as the inspiration.

iii.) Ubuntu (Linux Kernel):

• Type: Monolithic kernel with support for modules.

History : In the early 90s Linus Torvalds originally created Linux an open-source Unix alternative which has since become the kernel for most of the worlds Linux distributions.

5. Conclusion

The foreign bit I learnt is that it's like a brain for a computer Operating System. This includes everything that goes on inside the computer, so it is super important. I came to know about Kernels like monolithic which does all the work and micro kernels which are minimalistic. And I learned about hybrid kernels that combine these ideas.

It turned out for some famous kernels too. This is used by several operating systems such as iOS which uses XNU, Windows NT kernel, and Linux for Ubuntu. All of them tell their own tale and were created for a million different reasons. Some are focused on being secure, and others on working well both on phones as well as you laptop.

Eventually, I learned that having great kernels are key to how well our computers

6. References

Modern operating system third edition by Andrew S. Tanenbaum

Linux with operating sytem Concepts second edition by Richard Fox