Module Code & Module Title

Level 5 – CT5052NP Network Operating System

Assessment Type

Logbook Report

Semester

2023/24 Spring/Autumn

Student Name: Bishow Shrestha

London Met ID: 23048785

College ID: NP04CP4A230207

Assignment Due Date: 2024/09/30

Assignment Submission Date: 2024/09/28

Submitted To: Prashant Adhikari

Word Count (Where Required):1921

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.

TABLE OF CONTENTS

1.Introduction

Kernel is known as a bunch of codes which can also be referred as core of operating system which work as a main interface between hardware and software of a computer. Kernel gives access to different applications running on computer to use hardware resources like storage, CPU,I/O and networking.

It includes major system functions like process management, scheduling and memory management.

It also ensures the smooth interaction between software applications and hardware components.

2.Objective

In the context of operating systems, the known objective of Kernel is to act as a bridge between software and hardware. It also perform some tasks like resources management, process management , memory management , device management and other system.

Objective explanation

1)System Processes : Kernel is responsible for the creation of a process followed by the scheduling of the process and termination of the process.

2)Resources Management : Kernel does a job of allocating system resources like CPU , Memory to different programs running on the system.

3)Memory Allocation : The another objective of kernel is to manage the system's available memory which includes Virtual Memory, Physical Memory etc. for smooth processing.

4)Device Management : Kernel controls and also communicates with the hardware devices through device drivers to provide a smooth interface to the software.

3.Types of kernel

There are many types of kernel but we can divide those types into two based on there structure format and functionality . So, the types are:

- Structure based division : This division is done on the basis of how the kernel is structured and organized with its components .The types of structure based division of kernel are:

    i. Monolithic kernel : In this kernel ,all system services like process management , memory allocation , hardware management is done in a single block. It is complex and large but fast . for example Linux and Unix

    ii. Microkernel : This kernel is specified for essential functions only like communication between hardware and software that run in the kernel and all other functions will run in non-system level applications run(user space).It is smaller and easier compared to Monolithic kernel. For example : Minix , QNX.

    iii. Hybrid Kernel : As you say it by it's name ,It is the combination of the elements of both monolithic and micro kernel structures. The functions and programs run in both kernel and user space too. For example : Windows NT , macOS.

    iv. Exokernel : This kernel has the minimum control over the resources so it pushes the service to applications themselves. It is also small but complex. For example : MIT's Exokernel.

- Functionality based division : Unlike structure based kernels , this division is based on the functionalities the kernel provides and how the task like resource management and security are handled. The major types are:

i. General-purpose kernels : As you go by the name, It is designed or made for general use computers such as PC , Laptops to provide different features and support for the interface between software and hardware. It is often structured as monolithic and hybrid kernels. Examples : Linux , Windows NT .

ii. Real-time kernels : This kernel is designed to provide real time services like real time data processing which can be used in time-sensitive apps like robots, medical system , weather forecasting. Example : RT Linux ,VxWorks.

iii. Security-focused kernels : Like its name ,The function is also designed to ensure high level of security which includes user access, resource abstraction etc. It is often based on microkernel design . Example : Security-Enhanced Linux(SELinux).

This concludes that Structure based division mainly focuses on the design format or structure of the kernel which the functionality based division is done by looking the functional ability of a kernel to handle the tasks .

4. Popular Kernels and their history

The most popular operating systems like windows , IOS , Ubuntu are using hybrid or monolithic kernels like Windows is using Windows NT , IOS is using XNU and Ubuntu is using Linux kernel. There selection of kernel plays a critical role in their performance and security .

4.1 Windows NT

In the case of Windows OS, They are using Windows NT Kernel which is a hybrid kernel in term of structure. Windows NT was developed in early 1990s as Microsoft was trying to build a new and more powerful operating system designed for enterprise ecosystem.

Another Microsoft operating system is Windows NT (NT stands for New Technology), which is compatible with Windows 95 at a certain level, but a complete rewrite from scratch internally. It is a full 32-bit system. The lead designer for Windows NT was David Cutler, who was also one of the designers of the VAX VMS operating system, so some ideas from VMS are present in NT. In fact, so many ideas from VMS were present in it that the owner of VMS, DEC, sued Microsoft. The case was settled out of court for an amount of money requiring many digits to express. Microsoft expected that the first version of NT would kill off MS-DOS and all other versions of Windows since it was a vastly superior system, but it fizzled. Only with Windows NT 4.0 did it finally catch on in a big way, especially on corporate networks. Version 5 of Windows NT was renamed Windows 2000 in early 1999. It was intended to be the successor to both Windows 98 and Windows NT 4.0. (Tanenbaum, 2009)

Key points to know about Windows NT Kernel:

- A Developer named Dave Cutler was hired to lead the development.
- NT Kernel (Hybrid kernel) is a combination of both monolithic and microkernel structure.
- It’s hybrid approach allows it to take advantage of both speed and modularity.
- Windows continue to use this kernel due to its flexibility and robust feature set.

4.2 XNU

For IOS , They are using XNU(“X is Not Unix”) kernel which is also a hybrid kernel .XNU was originally developed by NeXT Operating System of NeXT Inc. after Apple chose for NeXT and announced the acquisition of NeXT on December 20, 1996.

Ever since the acquisition of NeXT, Apple continued working on their new operating system under the code name Rhapsody, which would eventually be released as Mac OS X (which was renamed to just OS X with the release of Mountain Lion). Before the consumer version was released, Apple first released Mac OS X Server 1.0 on March 16, 1999. This server OS contained components from all of the operating systems Apple owned at the time (Mac OS, NeXTSTEP and Mac OS X). The consumer edition, called Mac OS X 10.0, was released on March 24, 2001, more then four years after the acquisition of NeXT. Mac OS X was the first operating system by Apple that featured the XNU kernel, and it has been the OS X kernel ever since. Apple made some changes to the original XNU kernel over time, most notably the additions of I/O Kit for driver development, the update of the Mach component from version 2.5 to version 3.0 and the porting of XNU to the Intel and ARM processors as well as the transition from 32-bits to 64-bits. (Keuper, 2012)

4.3 Linux Kernel

Ubuntu is currently using Linux kernel which is a monolithic kernel which perform all task like process management , hardware management and memory management in a single block which is a complex work but is a fast way.

It has all the features you would expect in a modern fully-fledged Unix, including true multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables, proper memory management, and multistack networking including IPv4 and IPv6. Although originally developed first for 32-bit x86-based PCs (386 or higher), today Linux also runs on a multitude of other processor architectures, in both 32- and 64-bit variants. (Kernel, 2024)

In the 1960s, Massachusetts Institute of Technology (MIT) and a host of companies developed an experimental operating system called Multics (or Multiplexed Information and Computing Service) for the GE-645. One of the developers of this operating system, AT&T, dropped out of Multics and developed their own operating system in 1970 called Unics. Along with this operating system was the C language, for which C was developed and then rewritten to make operating system development portable. Twenty years later, Andrew Tanenbaum created a microkernel version of UNIX®, called MINIX (for minimal UNIX), that ran on small personal computers. This open source operating system inspired Linus Torvalds' initial development of Linux in the early 1990s (Jones, 2007)

5.Boot Process

Definition: Boot process is a series of steps that start to occur once after the computer is turned on and starts up , initializing the operating system and preparing the whole system for use.

The steps are:

- POST
- BIOS/UEFI Execution
- MBR or GPT
- Bootloader Execution
- Kernel
- OS initialization
- Login

EXPLAINATION

When the computer is booted, the BIOS is started. It first checks to see how much RAM is installed and whether the keyboard and other basic devices are installed and responding correctly. Typically, an attempt is made to boot from a CD-ROM (or sometimes USB) drive, if one is present. If that fails, the system boots from the hard disk. The first sector from the boot device is read into memory and executed. This sector contains a program that normally examines the partition table at the end of the boot sector to determine which partition is active. Then a secondary boot loader is read in from that partition. This loader reads in the operating system from the active partition and starts it. The operating system then queries the BIOS to get the configuration information. For each device, it checks to see if it has the device driver. Then it initializes its tables, creates whatever background processes are needed, and starts up a login program or GUI. (Tanenbaum, Boot Process, 2014)

6. Conclusion

Conclusively, The kernel is the core part of an operating system that is responsible for managing communication between software and hardware. It also handles other important tasks such as processing, memory management , device management etc. Different types of kernels such as monolithic, microkernel and hybrid have different structure and different way of functioning. Popular systems like Windows, macOS , and Linux use different kernels based on their needs. For example, Windows uses Windows NT kernel(hybrid kernel) while ubuntu uses Linux kernel(monolithic kernel). Overall, Kernel is essential for the smooth functioning of any computer system.

7. References

Jones, M. T. (2007). Anatomy of linux kernel.

Kernel. (2024, 08 06). Retrieved from Kernel.org: https://www.kernel.org/linux.html

Keuper, D. (2012). XNU : a security evaluation.

Tanenbaum, A. S. (2009). *Modern Operating System(3rd Edition).*

Tanenbaum, A. S. (2014). Boot Process. In *Modern Operating System (4th editon).*