MODULE 1 – Essentials of Operating Systems (OS)

ITRACKA1- Platform Technologies: Advanced Operating System

Disclaimer

This learning material is used in compliance with the flexible teaching-learning approach espoused by CHED in response to the pandemic that has globally affected educational institutions. Authors and publishers of the contents are well acknowledged. As such the college and its faculty do not claim ownership of all sourced information. This learning material will solely be used for instructional purposes not for commercialization.

College of Information and Communications Technology

Overview

In this module we will explore and be updated in the essentials of Operating Systems (OS). This module is composed of: *Overview* which provides a quick introduction to the module; *Learning Outcomes* which state the expected knowledge, behaviors and skills that the students must gain after the lessons presented ; *Pre-Test* which evaluates the prior knowledge of students in the topics to be presented ; *Definition of Key Terms/Unlocking of Difficulties*; the *Lesson Proper*; *Supplementary Learning Resources* which gives alternative or additional resources to students; *Activities* which enhances the retention and skills of students; and *Self-Assessment Questions* which serves as the evaluation how far the students gained knowledge or skills in the lessons presented.

Learning Outcomes

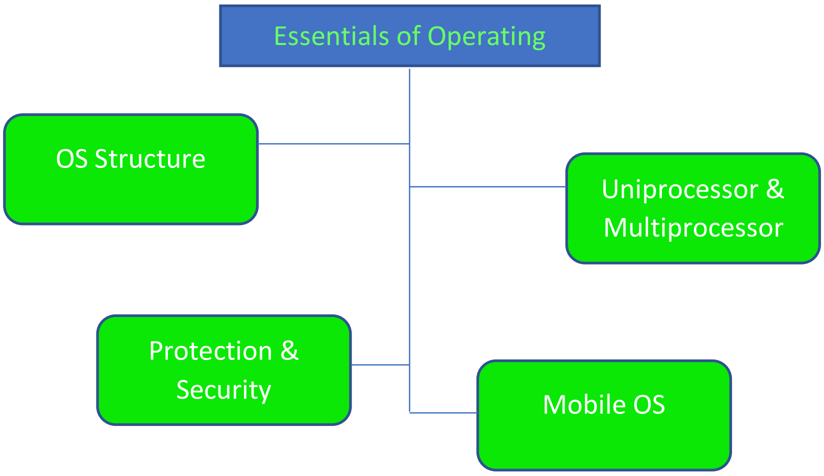
At the end of the lesson the students shall be able to:

1. determine the different operating system structures
2. differentiate uniprocessor and multiprocessor systems
3. identify the appropriate mechanism to protect and secure computer systems
4. identify the current mobile OS

Pre-Test

1. What are the components of an operating systems?
2. What are the types of operating systems?
3. What are the functions of operating system?
4. What is a kernel?

Module Map



Definition of Key Terms/Unlocking of Difficulties

Availability – refers to ensuring that authorized parties are able to access the information when needed.

Confidentiality –limits access to information to protect information from getting misused by any unauthorized access.

Direct Memory Access (DMA)– is a feature of computer systems that allows certain hardware subsystems to access main system memory (random-access memory) independent of the central processing unit (CPU).

Integrity – means the assurance of the accuracy, and completeness of data.

Introduction

Operating system is a program that serves as the interface between the hardware and software in a computer system. It takes control of the computer system once the booting process was completed. To easily understand the significance of the major features of an operating system it useful to be aware/recall and updated in the essentials of operating systems. Topics included in this module are: *OS Structure; Uniprocessor and Multiprocessor; Protection and Security; and Mobile OS*.

Lesson 1.1 - OS Structure

Operating system can be implemented with the help of various structures. The structure of the OS depends mainly on how the various common components of the operating system are interconnected and melded into the kernel. Depending on this we have following structures of the operating system:

Simple Structure

Such operating systems do not have well defined structure and are small, simple and limited systems. The interfaces and levels of functionality are not well separated. MS-DOS is an example of such operating system. In MS-DOS application programs are able to access the basic I/O routines. These types of operating system cause the entire system to crash if one of the user programs fails.

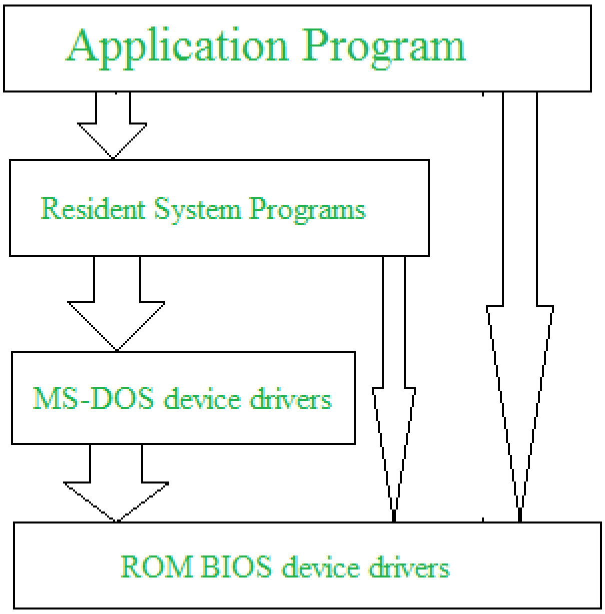
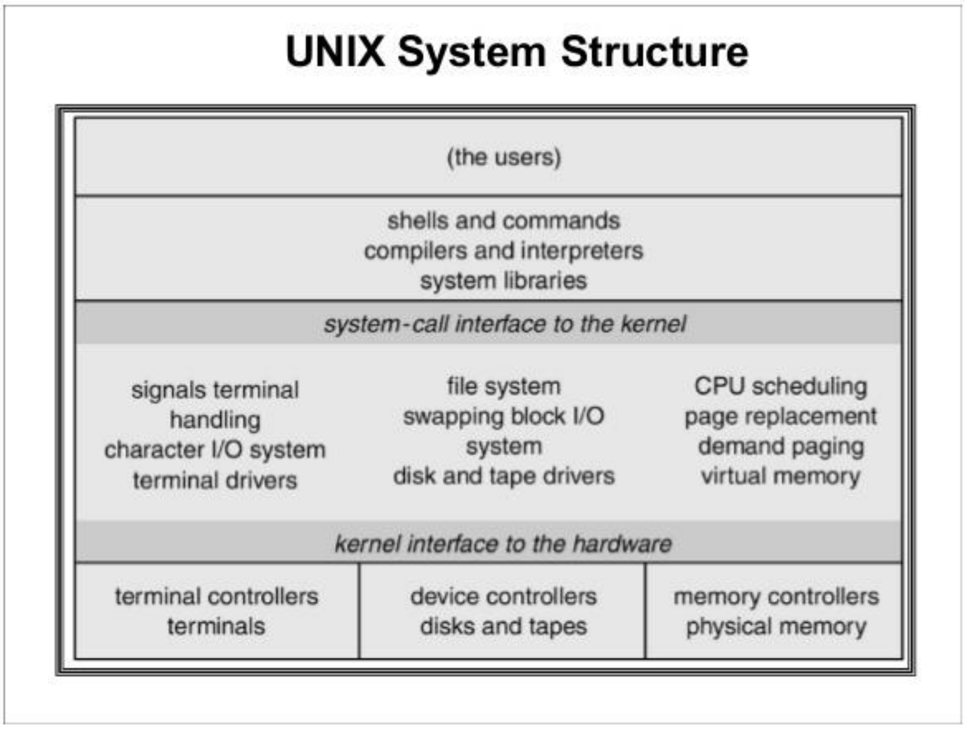


Figure 1.1a – MS-DOS layer Structure

The original UNIX OS used a simple layered approach, but almost all the OS was in one big layer, not really breaking the OS down into layered subsystem.



UNIX System Structure

(the users)

shells and commands
compilers and interpreters
system libraries

system-call interface to the kernel

signals terminal handling
character I/O system
terminal drivers

file system
swapping block I/O system
disk and tape drivers

CPU scheduling
page replacement
demand paging
virtual memory

kernel interface to the hardware

terminal controllers
terminals

device controllers
disks and tapes

memory controllers
physical memory

Figure 1.1b – Traditional UNIX system structure

Monolithic Approach

- A monolithic design of OS architecture makes no special accommodation for the special nature of the OS.
- Functionality of the OS is invoked with simple function calls within the kernel, which is one large program.
- Device drivers are loaded into the running kernel and become part of the kernel.
- The oldest architecture used for developing OS.

Layered Approach

- This approach breaks up the operating system into different layers.
- Each layer relies solely on the services provided by the next lower layer.
- It allows each layer to be developed and debugged independently.
- One of the drawbacks of this approach is that no layer can call upon the services of any higher layer.
- Example: OS/2, Windows NT(1st release)

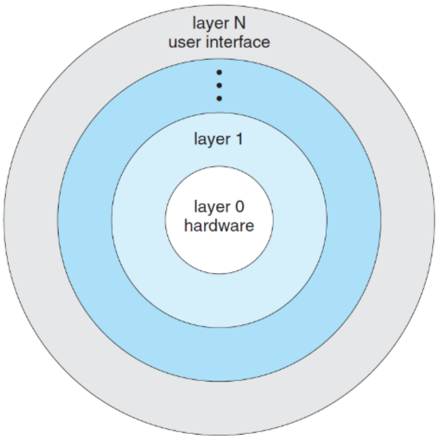


Figure 1.1c – Layered OS

Microkernels

- This OS architecture targets robustness. The privileges granted to the individual parts of the OS are restricted as much as possible and the communication between the parts relies on a specialized communication mechanism that enforce the privileges as necessary.

- This also remove all non-essential services from the kernel, and implement them as system application instead, thereby making the kernel as small and efficient as possible.
- Most microkernels provide basic process and memory management, and message passing between other services and not much more.
- Security and protection can be enhanced, as most services are performed in user mode, not kernel mode.
- System expansion can also be easier, because it only involves adding more system applications, not rebuilding a new kernel.
- Mach was the first and most widely known microkernel, and now forms a major component of Mac OSX.
- Another microkernel example is QNX, a real-time OS for embedded systems.

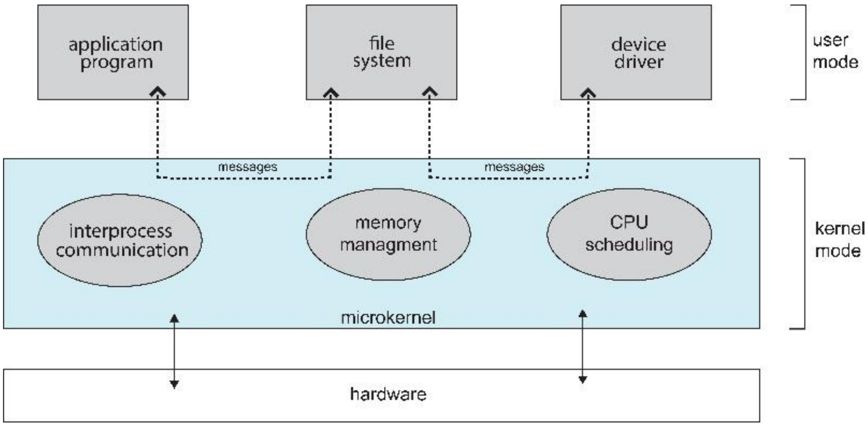


Figure 1.1d – Architecture of a typical microkernel

Other Structures

A. Modular Kernel

- Like monolithic layer structure but allows for partial loading running of various kernel components
- LINUX

B. Hybrid Structure

- Given that microkernel merely acts as a server for client requests, it is possible to provide support for more than one API at a time with a separate server for each, router through the microkernel.

- Most real operating systems are not pure examples of any of the operating system design models. Real systems are usually a mix of different design models.
- **MacOS and iOS MacOS and iOS systems** have a kernel environment known as Darwin. Darwin is an example of a hybrid system - a mix of different designs. It has a Mach micro-kernel and a BSD Unix kernel implemented over the Mach micro-kernel. Programs can make Mach system calls, and they can also make BSD system calls.

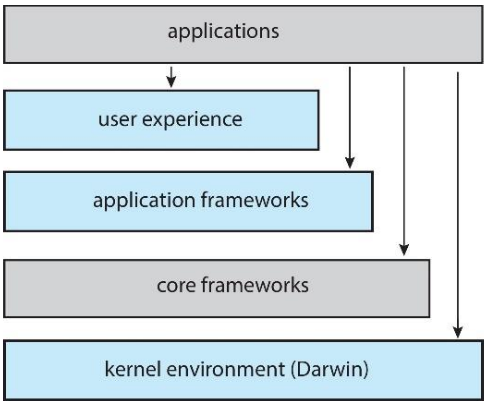


Figure 2.16: Architecture of Apple's macOS and iOS operating systems

- Android systems have a modified Linux kernel

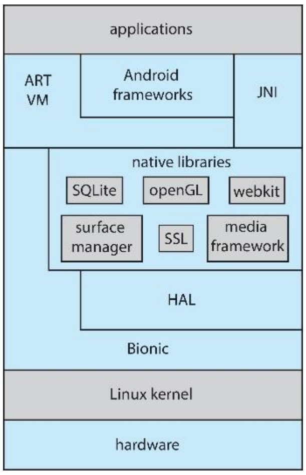


Figure 2.18: Architecture of Google's Android

- Windows Subsystem for Linux Windows has some subsystems that emulate other operating systems. For example, there is a subsystem for Linux that allows users to run native Linux applications on a Windows machine. When an application running in the Linux environment makes a system call, it is (basically) translated into Windows system calls.

Lesson 1.2 - Uniprocessor and Multiprocessor

According to 3G E-learning LLC, a computer system can be organized in a number of different ways, which we can categorize roughly according to the number of general-purpose processors used.

Single-Processor Systems

Until recently, most computer systems used a single processor. On a single processor system, there is one main CPU capable of executing a general-purpose instruction set, including instructions from user processes.

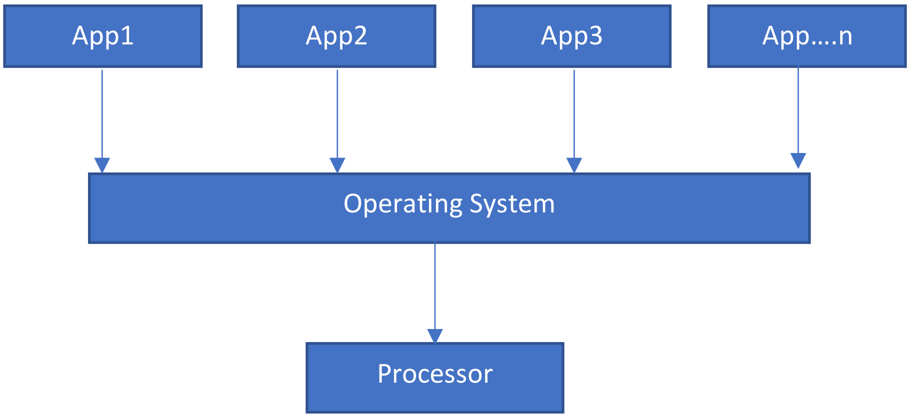


Figure 1.2a - Single-Processor System

Differences Between Single Processor and Multiprocessor Systems

There are many differences between single processor and multiprocessor systems. Some of these are illustrated as follows:

- Single processor system contains only one processor while multiprocessor systems may contain two or more processors.
- Single processor systems use different controllers for completing special tasks such as DMA (Direct Memory Access) Controller. On the other hand, multiprocessor systems have many processors that

can perform different tasks. This can be done in symmetric or asymmetric multiprocessing.

- Single processor systems can be more expensive than multiprocessor systems. If n processor multiprocessor system is available, it is cheaper than n different single processor systems because the memory, peripherals etc. are shared.
- It is easier to design a single processor system as compared to a multiprocessor system. This is because all the processors in the multiprocessor system need to be synchronized and this can be quite complicated.
- Throughput of a multiprocessor system is more than a single processor system. However, if the throughput of n single processor systems is T then the throughput of n processor multiprocessor system will be less than T.
- Single processor systems are less reliable than multiprocessor systems because if the processor fails for some reason then system cannot work. In multiprocessor systems, even if one processor fails than the rest of the processors can pick up the slack. At most the throughput of the system decreases a little.
- Most modern personal computers are single processor systems while multiprocessors are used in niche systems only.

Multiprocessor Systems

Within the past several years, multiprocessor systems (also known as parallel systems or multicore systems) have begun to dominate the landscape of computing. Such systems have two or more processors in close communication, sharing the computer bus and sometimes the clock, memory, and peripheral devices. Multiprocessor systems first appeared prominently in servers and migrated to desktop and laptop systems. Recently, multiple processors have appeared in mobile devices such as smartphones and tablet computers.

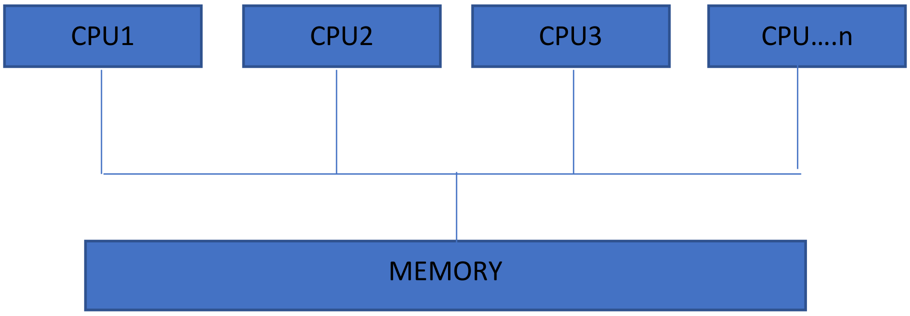


Figure 1.2b – Multiprocessor Architecture

Types of Multiprocessors

There are mainly two types of multiprocessors i.e. symmetric and asymmetric multiprocessors.

Symmetric Multiprocessors

In these types of systems, each processor contains a similar copy of the operating system and they all communicate with each other. All the processors are in a peer to peer relationship i.e. no master - slave relationship exists between them.

An example of the symmetric multiprocessing system is the Encore version of Unix for the Multimax Computer.

Asymmetric Multiprocessors

In asymmetric systems, each processor is given a predefined task. There is a master processor that gives instruction to all the other processors. Asymmetric multiprocessor system contains a master slave relationship.

Asymmetric multiprocessor was the only type of multiprocessor available before symmetric multiprocessors were created. Now also, this is the cheaper option.

Advantages of Multiprocessor Systems

1. More reliable Systems

In a multiprocessor system, even if one processor fails, the system will not halt. This ability to continue working despite hardware failure is known as graceful degradation. For example: If there are 5 processors in a multiprocessor system and one of them fails, then also 4 processors are still working. So the system only becomes slower and does not ground to a halt.

2. Enhanced Throughput

If multiple processors are working in tandem, then the throughput of the system increases i.e. number of processes getting executed per unit of time increase. If there are N processors then the throughput increases by an amount just under N.

3. More Economic Systems

Multiprocessor systems are cheaper than single processor systems in the long run because they share the data storage, peripheral devices, power supplies etc. If there are multiple processes that share data, it is better to schedule them on multiprocessor systems with shared data than have different computer systems with multiple copies of the data.

Disadvantages of Multiprocessor Systems

1. Increased Expense

Even though multiprocessor systems are cheaper in the long run than using multiple computer systems, still they are quite expensive. It is much cheaper to buy a simple single processor system than a multiprocessor system.

2. Complicated Operating System Required

There are multiple processors in a multiprocessor system that share peripherals, memory etc. So, it is much more complicated to schedule processes and impart resources to processes than in single processor systems. Hence, a more complex and complicated operating system is required in multiprocessor systems.

3. Large Main Memory Required

All the processors in the multiprocessor system share the memory. So a much larger pool of memory is required as compared to single processor systems.

Clustered Systems

Clustered systems are similar to parallel systems as they both have multiple CPUs. However a major difference is that clustered systems are created by two or more individual computer systems merged together. Basically, they have independent computer systems with a common storage and the systems work together.

The clustered systems are a combination of hardware clusters and software clusters. The hardware clusters help in sharing of high-performance disks between the systems. The software clusters make all the systems work together .

Each node in the clustered systems contains the cluster software. This software monitors the cluster system and makes sure it is working as required. If any one of the nodes in the clustered system fail, then the rest of the nodes take control of its storage and resources and try to restart.

Types of Clustered Systems

There are primarily two types of clustered systems i.e. asymmetric clustering system and symmetric clustering system. Details about these are given as follows −

Asymmetric Clustering System

In this system, one of the nodes in the clustered system is in hot standby mode and all the others run the required applications. The hot standby mode is a failsafe in which a hot standby node is part of the system

. The hot standby node continuously monitors the server and if it fails, the hot standby node takes its place.

Symmetric Clustering System

In symmetric clustering system two or more nodes all run applications as well as monitor each other. This is more efficient than asymmetric system as it uses all the hardware and doesn't keep a node merely as a hot standby.

Attributes of Clustered Systems

There are many different purposes that a clustered system can be used for. Some of these can be scientific calculations, web support etc. The clustering systems that embody some major attributes are −

- Load Balancing Clusters

  In this type of clusters, the nodes in the system share the workload to provide a better performance. For example: A web based cluster may assign different web queries to different nodes so that the system performance is optimized. Some clustered systems use a round robin mechanism to assign requests to different nodes in the system.

- High Availability Clusters

  These clusters improve the availability of the clustered system. They have extra nodes which are only used if some of the system components fail. So, high availability clusters remove single points of failure i.e. nodes whose failure leads to the failure of the system. These types of clusters are also known as failover clusters or HA clusters.

Benefits of Clustered Systems

The difference benefits of clustered systems are as follows −

- Performance

  Clustered systems result in high performance as they contain two or more individual computer systems merged together. These work as a parallel unit and result in much better performance for the system.

- Fault Tolerance

  Clustered systems are quite fault tolerant and the loss of one node does not result in the loss of the system. They may even contain one or more nodes in hot standby mode which allows them to take the place of failed nodes.

- Scalability

  Clustered systems are quite scalable as it is easy to add a new node to the system. There is no need to take the entire cluster down to add a new node.

Lesson 1.3 - Protection and Security

Computer resources requires to be free from attacks. Ensuring the protection and security of these resources is one of the operating system functions. Protection and security of the system maintain the availability, confidentiality, and integrity of services provided.

Common risks in the system:

A. People Risks

By some estimates, human errors, ignorance, and omissions cause more than half of all security breaches sustained by networks. Human error accounts for so many security breaches because taking advantage of people is often an easy way to circumvent network security. End-user awareness and training can be a monumental task that requires regular attention and due diligence.

Common types of social engineering include the following:

- Phishing - An electronic communication that appears to come from a legitimate person or organization and requests access or authentication information.
- Baiting - A malware-infected file, such as a free music download, or device, such as a USB flash drive, is seemingly left unguarded for someone to take and attempt to use on their own computer.
- Quid Pro Quo - A gift or service is offered in exchange for private information or “temporary” access to the user’s computer system. This tactic is surprisingly effective with employees who have not been adequately trained to detect social engineering attempts.
- Tailgating - A person posing as an employee or a delivery or service provider follows an authorized employee into a restricted area.

B. Technology Risks

Technology risks inherent in all seven layers of the OSI model.

The following risks are inherent in network hardware and design:

a. Spoofing Attack - MAC addresses can be impersonated in an attack called spoofing. Other types of spoofing attacks involve impersonating IP addresses. IP address spoofing can result in DoS (denial of service) attacks or modified DNS messages.
b. DoS (denial of service) attack - this attack occurs when a legitimate user is unable to access normal network resources, such as a web server, because of an attacker's intervention. Most often, this type of attack is achieved by flooding a system with so many requests for services that it cannot respond to any of them, as a result, all data transmissions are disrupted.

C. Malware Risks

Malware (short for malicious software) refers to any program or piece of code designed to intrude upon or harm a system or its resources.

Types of Malware

- Virus - program that replicates itself with the intent to infect more computers, either through network connections when it piggybacks on other files or through the exchange of external storage devices. A virus might damage files or systems, or it might simply annoy users by flashing messages or pictures on the screen.
- Trojan horse (or Trojan) - program that disguises itself as something useful but harms your system; named after the famous wooden horse in which soldiers were hidden. Because Trojan horses do not replicate themselves, they are not considered viruses.
- Worm - program that runs independently of other software and travels between computers and across networks. They may be transmitted by any type of file transfer, including email attachments. Worms do not alter other programs in the same way that viruses do, but they can carry viruses.
- Bot (short for robot) - process that runs automatically, without requiring a person to start or stop it. Bots can be beneficial or malicious. Especially when used for ill intent, it does not require user interaction to run or propagate itself.

Instead, it connects to a central server (called a command-and-control server, or C&C server) which then commands an entire botnet of similarly infected devices. Bots can be used to damage or destroy a computer's data or system files, issue objectionable content, launch DoS attacks, or open back doors for further infestation.

- Ransomware - program that locks a user's data or computer system until a ransom is paid. In most cases, the infection encrypts data on the computer, and can also encrypt data on backup devices, removable storage devices, and even cloud storage accounts connected to the computer, such as Dropbox or OneDrive. Currently, the only mostly reliable defense is to make manual backups of data on a regular basis and disconnect the backup media from the computer between backups.

Protection and Security Methods

- Authentication - user authentication, is the process of verifying a user's credentials (typically a username and password) to grant the user access to secured resources on a system or network. In other words, authentication asks the question, "Who are you?"

- Authorization - once a user has access to the network, the authorization process determines what the user can and cannot do with network resources. In other words, authorization asks the question, "What are you allowed to do?" Authorization restrictions affect Layer 2 segmentation, Layer 3 filtering, and Layer 7 entitlements. For example, what VLAN are you assigned to? What servers or databases can you access? What commands can you run on a device?

- Accounting - the accounting system logs users' access and activities on the network. In other words, accounting asks, "What did you do?" The records that are kept in these logs are later audited, either internally or by an outside entity, to ensure compliance with existing organizational rules or external laws and requirements.

Lesson 1.4 - Mobile OS

A mobile operating system, also called a mobile OS, is an operating system that is specifically designed to run on mobile devices such as mobile phones, smartphones, PDAs, tablet computers and other handheld devices. The operating system is responsible for determining the functions and features available on your device, such as thumb wheel, keyboards, WAP, synchronization with applications, email, text messaging and more. The mobile OS will also determine which third-party applications (mobile apps) can be used on your device.

Popular Mobile Operating Systems

1. Android OS (Google Inc.)

   The Android mobile operating system is Google's open and free software stack that includes an operating system, middleware and also key applications for use on mobile devices, including smartphones. Updates for the open source Android mobile operating system have been developed under "dessert-inspired" version names (Cupcake, Donut, Eclair, Gingerbread, Honeycomb, Ice Cream Sandwich) with each new version arriving in alphabetical order with new enhancements and improvements.

2. Bada (Samsung Electronics)

   Bada is a proprietary Samsung mobile OS that was first launched in 2010. The Samsung Wave was the first smartphone to use this mobile OS. Bada provides mobile features such as multipoint-touch, 3D graphics and of course, application downloads and installation.

3. BlackBerry OS (Research In Motion)

   The BlackBerry OS is a proprietary mobile operating system developed by Research In Motion for use on the company’s popular BlackBerry handheld devices. The BlackBerry platform is popular with corporate users as it offers synchronization with Microsoft Exchange, Lotus Domino, Novell GroupWise email and other business software, when used with the BlackBerry Enterprise Server.

4. iPhone OS / iOS (Apple)

   Apple's iPhone OS was originally developed for use on its iPhone devices. Now, the mobile operating system is referred to as iOS and is supported on several Apple devices including the iPhone, iPad, iPad 2 and iPod Touch. The iOS mobile operating system is available only on Apple's own manufactured devices as the company does not license the OS for third-party hardware. Apple iOS is derived from Apple's Mac OS X operating system.

5. MeeGo OS (Nokia and Intel)

A joint open source mobile operating system which is the result of merging two products based on open source technologies: Maemo (Nokia) and Moblin (Intel). MeeGo is a mobile OS designed to work on a number of devices including smartphones, netbooks, tablets, in-vehicle information systems and various devices using Intel Atom and ARMv7 architectures.

6. Palm OS (Garnet OS)

The Palm OS is a proprietary mobile operating system (PDA operating system) that was originally released in 1996 on the Pilot 1000 handheld. Newer versions of the Palm OS have added support for expansion ports, new processors, external memory cards, improved security and support for ARM processors and smartphones. Palm OS 5 was extended to provide support for a broad range of screen resolutions, wireless connections and enhanced multimedia capabilities and is called Garnet OS.

7. Symbian OS (Nokia)

Symbian is a mobile operating system (OS) targeted at mobile phones that offers a high-level of integration with communication and personal information management (PIM) functionality. Symbian OS combines middleware with wireless communications through an integrated mailbox and the integration of Java and PIM functionality (agenda and contacts). Nokia has made the Symbian platform available under an alternative, open and direct model, to work with some OEMs and the small community of platform development collaborators. Nokia does not maintain Symbian as an open source development project.

8. webOS (Palm/HP)

WebOS is a mobile operating system that runs on the Linux kernel. WebOS was initially developed by Palm as the successor to its Palm OS mobile operating system. It is a proprietary Mobile OS which was eventually acquired by HP and now referred to as webOS (lower-case w) in HP literature. HP uses webOS in a number of devices including several smartphones and HP TouchPads. HP has pushed its webOS into the enterprise mobile market by focusing on improving security features and management with the release of webOS 3.x. HP has also announced plans for a version of webOS to run within the Microsoft Windows operating system and to be installed on all HP desktop and notebook computers in 2012.

9. Windows Mobile (Windows Phone)

Windows Mobile is Microsoft's mobile operating system used in smartphones and mobile devices – with or without touchscreens. The Mobile OS is based on the Windows CE 5.2 kernel. In 2010 Microsoft announced a new smartphone platform called Windows Phone 7.

Android versions

Codenames are used to describe the various updates for the open source Android mobile operating system. Android versions are developed under dessert-inspired codenames (up until the 2019 Android 10 release), with each new version arriving in alphabetical order with new enhancements and improvements to the Android SDK.

No Codename (v1.0) - Debuted in Fall 2008: Google Android 1.0 officially arrived September 23rd, 2008, as the only release of Android (so far) to not include a codename

Petit Four (v1.1) - Debuted in February 2009: Android 1.1 "Petit Four" began rolling out in early 2009 as the first update for the new Google Android mobile operating system.

Cupcake (v1.5) - Debuted in April 2009: Key additions: Speech recognition tools, a virtual keyboard, video upload support for YouTube and support for live data feeds and live folders.

Donut (v1.6) - Debuted in Fall 2009: Key additions: Support for CDMA smartphones, additional screen sizes and a text-to-speech engine.

Eclair (v2.0) - Debuted in October 2009: Key additions: Support for multi-touch devices, new browser interface, Microsoft Exchange support, single interface for managing multiple online accounts, soft keys support, and an enhanced camera app (with digital zoom and flash support).

FroYo (v2.2) - Debuted in Fall 2010: Key additions: USB tethering support (for turning a smartphone into a Wi-Fi hotspot), significant speed improvements, Flash 10.1 support, voice dialing over Bluetooth, the ability to store apps on external memory cards, updated browser with Google Chrome's V8 JavaScript.

Gingerbread (v2.3) - Debuted in December 2010: Key additions: Google Voice over Wi-Fi, enhanced gaming functionality, improved Google Apps.

Honeycomb (v3.0) - Debuted in February 2011: Key additions: A tablet-centric update that delivered a new interface optimized for devices with larger screen sizes (particularly tablets), video chat support based on Google Talk protocols, new System Bar for global status and notifications and Action Bar for application control, tabbed Web browsing, optimized soft keyboard and a new email interface.

Ice Cream Sandwich (v4.0) – Debuted in October 2011: Key additions: A smartphone-centric update based on the Linux kernel v3.0.1 that brings many of Honeycomb's features to smartphones, including Face Unlock facial recognition software, tabbed Web browsing capabilities, unified social networking contacts, 1080p video recording capabilities and video chat support based on Google Talk protocols.

Jelly Bean (v4.1, v4.2 and v4.3) – Debuted in June 2012: Key additions: Advanced natural language voice command capabilities akin to Apple's Siri, enhanced interface and overall responsiveness via "Project Butter," Google Now support, an improved Web browser, enhanced file management capabilities and more.

KitKat (v4.4) - Debuted in November 2013. Key additions: Full-screen immersive mode, new transitions framework, and "Project Svelte," a project initiated to reduce the memory needs of the Android OS. Originally internally referred to as Key Lime Pie, Google announced in early September 2013 that it would be using the iconic candy bar as the codename for the 4.4 Android release.

Lollipop (v5.0) - Debuted in November 2014. Key additions: Enhanced Material Design user interface, improved continuity across Android devices, multiple user support, a guest user account option, a new notification system, support for 64-bit CPUs, and more.

Marshmallow ("M Release") (v6.0) - Debuted in November 2015. Key additions: Now on Tap functionality and other Google Now enhancements, native fingerprint authentication support, Android Pay integration, USB Type-C support, improved battery life, better app management and more.

Oreo ("O Release") (v8.0) - Debuted in August 2017. Key additions: Android Go minimized version of Android mobile OS, picture-in-picture video support, Notification Grouping, Bluetooth 5 support, Wi-Fi Aware feature, and more.

Pie ("P Release") (v9.0) - Debuted in August 2018. Key additions: Adaptive Battery, Digital Well-being Dashboard (monitors screen time usage), App Slices, Intuitive Gestures, Adaptive Brightness, and more.

Android 10 ("Q Release") (v10.0) - Debuted in September 2019. Key additions: Live Captioning, Smart Reply, Dark Mode, Gesture Navigation, Focus Mode, Family Link, and automatic ongoing security patches and updates.

Supplementary Learning Resources

- https://www.tutorialspoint.com/Multiprocessor-Systems
- https://www.slideshare.net/smileybemust/operating-system-structure?next_slideshow=1

References

Barnes (2017), *Operating Systems and Practice*, Larsen and Keller Education,New York USA

Albano, GM (2016). *Operating Systems*, PUP Open University, Sta Mesa, Manila

Killings et.al. (2019),*Core Concepts in Computer Science: Operating Systems*, 3G E-Learning LLC, USA

Irtegov.D(2014), *Operating Systems Fundamentals*, Laxmi Publication Ltd., New Delhi

Godbole, S. & Kahate, A. (2011). *Operating Systems Third Edition*. Tata McGraw Hill Education Private Limited, New Delhi

West, J. et.al.(2019), *Network+ Guide to Networks Eight Edition*. Cengage, 20 Channel Center Street Boston, MA, 02210 USA