



Subject:	Unix Lab(SE ITL402)		
Class:	SE IT / Semester – IV (CBCGS) / Academic year: 2017-18		
Name of Student:	Kazi Jawwad A Rahim		
Roll No:	28	Date of performance (DOP) :	
Assignment/Experiment No:	01	Date of checking (DOC) :	
Title: To study History and Architecture of Unix/Linux operating systems			
Marks:		Teacher's Signature:	

1. Aim: To study History and Architecture of Unix/Linux operating systems.

2. Prerequisites:

C Programming Language and Operating System

3. Hardware Requirements:

- PC with minimum 2GB RAM

4. Software Requirements:

- Fedora installed.

5. Learning Objectives:

To study History and Architecture of Unix/Linux operating systems

6.Course Objectives Applicable: LO1

7. Program Outcomes Applicable: PO1

8. Program Education Objectives Applicable: PEO1

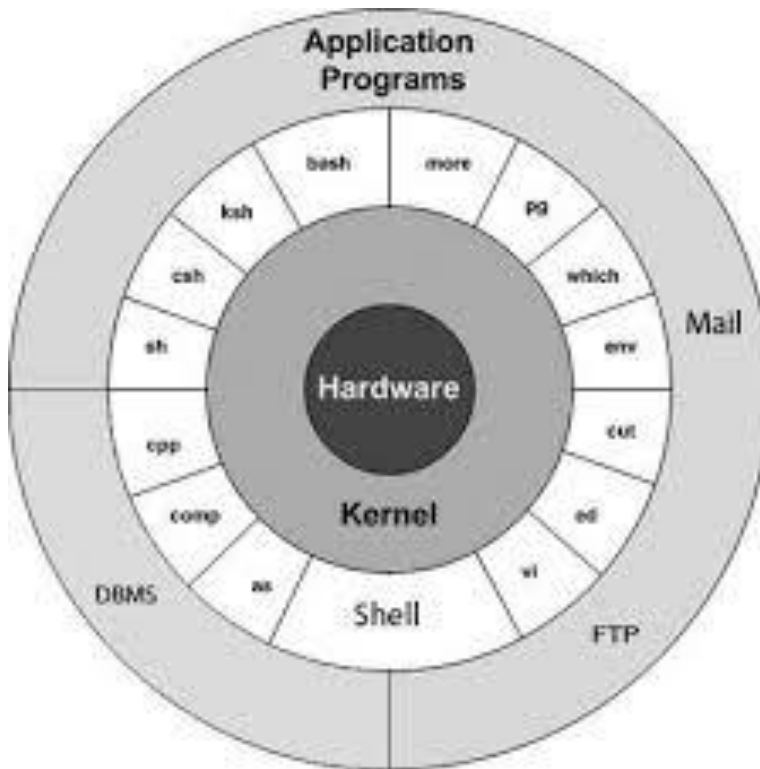
9. Theory:

History of Unix

The pre-history of Unix dates back to the mid-1960s when the Massachusetts Institute of Technology, Bell Labs, and General Electric were developing an innovative time-sharing operating system called Multics for the GE-645 mainframe. Multics introduced many innovations, but had many problems. Frustrated by the size and complexity of Multics but not by the aims, Bell Labs slowly pulled out of the project. Their last researchers to leave Multics, Ken Thompson, Dennis Ritchie, M. D. McIlroy, and J. F. Ossanna, decided to redo the work on a much smaller scale. The new operating system was initially without organizational backing, and also without a name. At this stage, the new operating system was a single-tasking operating system, not a multitasking one such as Multics. The name Unics (Uniplexed Information and Computing Service, pronounced as "eunuchs"), a pun on Multics (Multiplexed Information and Computer Services), was initially suggested for the project in 1970. Brian Kernighan claims the coining for himself, and adds that "no one can remember" who came up with the final spelling Unix. Dennis Ritchie, Doug McIlroy, and Peter G. Neumann also credit Kernighan. In 1972, Unix was rewritten in the C programming language. The migration from assembly to the higher-level language C resulted in much more portable software, requiring only a relatively small amount of machine-dependent code to be replaced when porting Unix to other computing platforms. Bell Labs produced several versions of Unix that are collectively referred to as Research Unix. In 1975, the first source license for UNIX was sold to Donald B. Gillies at the University of Illinois Department of Computer Science. UIUC graduate student Greg Cheson (who had worked on the UNIX kernel at Bell Labs) was instrumental in negotiating the terms of this license. During the late 1970s and early 1980s, the influence of Unix in academic circles led to large-scale adoption of Unix (BSD and System V) by commercial startups, including Sequent, HP-UX, Solaris, AIX, and Xenix. In the late 1980s, AT&T Unix System Laboratories and Sun Microsystems developed System V Release 4 (SVR4), which was subsequently adopted by many commercial Unix vendors. In the 1990s, Unix-like systems grew in popularity as Linux and BSD distributions were developed through collaboration by a worldwide network of programmers. In 2000, Apple released Darwin, also a Unix-like system, which became the core of the Mac OS X operating system, later macOS. Unix operating systems are widely used in modern servers, workstations, and mobile devices.

Unix architecture

A Unix architecture is a computer operating system system architecture that embodies the Unix philosophy. It may adhere to standards such as the Single UNIX Specification (SUS) or similar POSIX IEEE standard. No single published standard describes all Unix architecture computer operating systems - this is in part a legacy of the Unix wars.



Description: There are many systems which are Unix-like in their architecture. Notable among these are the GNU/Linux distributions. The distinctions between Unix and Unix-like systems have been the subject of heated legal battles, and the holders of the UNIX brand, The Open Group, object to "Unix-like" and similar terms. For distinctions between SUS branded UNIX architectures and other similar architectures, see Unix-like.

Kernel: A Unix kernel — the core or key components of the operating

system — consists of many kernel subsystems like process management, scheduling, file management, device management and network management, memory management, dealing with interrupts from hardware devices.

Each of the subsystems has some features:

Concurrency: As Unix is a multiprocessing OS, many processes run concurrently to improve the performance of the system.

Virtual memory (VM): Memory management subsystem implements the virtual memory concept and users need not worry about the executable program size and the RAM size. Paging: It is a technique to minimize the internal as well as the external fragmentation in the physical memory.

Virtual file system (VFS): A VFS is a file system used to help the user to hide the different file systems complexities. A user can use the same standard file system related calls to access different file systems.

The kernel provides these and other basic services: interrupt and trap handling, separation between user and system space, system calls, scheduling, timer and clock handling, file descriptor management.

Features: Some key features of the Unix architecture concept are:

Unix systems use a centralized operating system kernel which manages system and process activities.

All non-kernel software is organized into separate, kernel-managed processes.

Unix systems are preemptively multitasking: multiple processes can run at the same time, or within small time slices and nearly at the same time, and any process can be interrupted and moved out of execution by the kernel. This is known as thread management.

Files are stored on disk in a hierarchical file system, with a single top location throughout the system (root, or "/"), with both files and directories, subdirectories, sub-subdirectories, and so on below it.

With few exceptions, devices and some types of communications between processes are managed and visible as files or pseudo-files within the file system hierarchy. This is known as everything is a file. However, Linus Torvalds states that this is inaccurate and may be better rephrased as "everything is a stream of bytes".

The UNIX operating system supports the following features and capabilities:

Multitasking and multiuser

Programming interface

Use of files as abstractions of devices and other objects

Built-in networking (TCP/IP is standard)

Persistent system service processes called "daemons" and managed by init or inet

Criticism

The UNIX-HATERS Handbook covers some of these design features as failures from the user point of view. However, although some information is quite dated and cannot be applied to modern Unixes such as Linux, Eric S. Raymond discovered that several issues are still prevailing, while others were resolved. Raymond concludes that not all concepts behind Unix can be deemed as nonfunctional even though the book's intention may have been to portray Unix as inferior without encouraging discussions with developers to actually fix the issues.

11. Learning Outcomes Achieved

Students will be able to identify the basic Unix general purpose commands

12. Conclusion:

UNIX has a deep history which helps understand its effective functionality and architecture.

13. Experiment/Assignment Evaluation

SR	Parameters	Weight	Excellent	Good	Average	Poor	Not as per requirement
		Scale Factor ->	5	4	3	2	0
1	Technical Understanding	25					
2	Performance / Execution	25					
3	Question Answers	20					
4	Punctuality	20					
5	Presentation	10					
	Total out of 100 --> #(to be converted as per term-work evaluation applicable to the subject)		$\Sigma (\text{Weight} * \text{Scale Factor})/5 = \underline{\hspace{2cm}}$				

References:

- [1] https://en.wikipedia.org/wiki/Unix_architecture
- [2] <https://en.wikipedia.org/wiki/Unix>
- [3] <https://www.tutorialspoint.com/unix/unix-getting-started.htm>

Viva Questions

- When was UNIX evolved?
- What is the architecture of UNIX?